НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І
ПРИРОКОРИСТУВАННЯ УКРАЇНИ

Факультет інформаційних технологій

Кафедра комп'ютерних систем, мереж та кібербезпеки

Лахно В. А., Гусєв Б. С.,
Касаткін Д. Ю., Хорольська К. В.

ЗАХИСТ ІНФОРМАЦІЇ В КОМП'ЮТЕРНИХ СИСТЕМАХ І КІБЕРБЕЗПЕКА

ЧАСТИНА 1

«МОДЕЛІ І МЕТОДИ ЗАХИСТУ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНОГО СЕРЕДОВИЩА НА ОСНОВІ ІНТЕЛЕКТУАЛЬНОГО РОЗПІЗНАВАННЯ ЗАГРОЗ»

(навчальний посібник для самостійної роботи студентів ОС Магістр спеціальностей 123 - Компютерна інженерія: ОПП – КсіМ, ОО – КСЗІ)

Київ - 2023

УДК 621.3.049.77.001.63

Л79

Рекомендовано до друку Вченою радою НУБіП України (протокол № 4 від 25 жовтня 2023 р.)

Рецензенти:

Терейковський І.А., д.т.н., професор, професор кафедри СПіСКС, факультет прикладної математики «КПІ» ім. І.Сікорського;
Криворучко О.В., д.т.н., професор, завідувач кафедри інженерії програмого забезпечення та кібербезпеки Державний торговельно-економічний університет;
Смірнов О.А., д.т.н., професор, завідувач кафедри кібербезпеки та програмного забезпечення, Центральноукраїнський національний технічний університет.

Л79 Захист інформації в комп'ютерних системах і кібербезпека, (Частина 1). Навчальний посібник / **Лахно В.А., Гусєв Б.С., Касаткін Д.Ю., Хорольська К.В.** // – Київ: Редакційно-видавничий відділ НУБіП України, 2023. – 301 с.

ISBN 978-617-8351-97-7

В навчальному посібнику представлені сучасні моделі і методи захисту інформаційно-комунікаційного середовища на основі інтелектуального розпізнавання загроз. Також наведені основні типи запам'ятовуючих пристроїв та інтегральні схеми зі структурою, що програмується. Виявлення кібератак є дуже актуальним завданням, деяким аспектам цієї складної проблеми й присвячено цей навчальний посібник

Навчальний посібник буде корисний студентам та читатчам, які вивчають дисципліни пов'язані зі створенням та експлуатацією цифрових засобів обробки інформації або цікавляться сучасними методами захисту інформації від кіберзагроз.

Призначено для студентів ОС «Магістр» спеціальностей 123 - Комп'ютерна інженерія: ОПП – Комп'ютерні системи і мережі, ОПП – Комп'ютерні системи захисту інформації.

ПЕРЕЛІК СКОРОЧЕНЬ

АС – автоматизована система

АІС – автоматизована інформаційна система

АРМ – автоматизоване робоче місце

АСУ – автоматизована система управління

ГКС – глобальна комп’ютерна система

ДТЗС – допоміжні технічні засоби та системи

ДПРЗ – дискретні процедури розпізнавання загроз

ЗЗІ – засоби захисту інформації

ІАЦ – інформаційно-аналітичний центр

ІБ – інформаційна безпека

ІКС – інформаційно-комунікаційна система

ІР – інформаційні ресурси

КВКС - критично важлива комп'ютерна система

КІС – корпоративна інформаційна система

КМ – комп’ютерна мережа

КНІ – комп’ютерний напад на інформацію

КСЗІ – комплексні системи захисту інформації

ЛВ – локальний вузол

МЗК – мережі загального користування

ММ – мультимножина

ОБС – обчислювальна система

ОЦ – обчислювальний центр

ОС – операційні системи

ПЗ – програмне забезпечення

ПІБ – політика інформаційної безпеки

ППР – показник поточних ризиків

СВАП – системи виявлення та протидії кібератакам

СІБ – система інформаційної безпеки

СЗІ – система захисту інформації

ТЗОІ – технічні засоби обробки інформації

ШІ – штучний інтелект

ЗМІСТ

ВСТУП

Інформаційна сфера стала сьогодні базою для розвитку всіх інших сфер у житті людини, суспільства та держави. В інформаційній сфері відбуваються різні події та явища, аналіз яких стає життєво необхідним для будь-якого об'єкта. У сучасних умовах все більше поширюється аксіома, що кіберзахист інформаційних технологій має за своїми характеристиками відповідати масштабам загроз та ризиків. Відхилення від цього правила призведе до додаткових збитків. Для кожної інформаційної системи (ІС) повинен бути оптимальний рівень кіберзахищеності, який необхідно постійно підтримувати. Немає сумнівів, кіберзахист дуже важливий для ІС. Однак немає відповіді на дуже важливе питання – наскільки рішення, які пропонуються та/або реалізуються, справді відповідають вимогам кіберзахисту ІС. Слід враховувати, що система кіберзахисту ІС повинна мати цільове призначення. В останні роки кібератаки (КА) стали широко застосовуватися не лише окремими хакерами чи об'єднаними в групи, а й державами структурами деяких країн, зокрема росії. Тому, виявлення кібератак є дуже актуальним завданням. Деяким аспектам цієї складної проблеми й присвячено цей навчальний посібник.

РОЗДІЛ 1.
СУЧАСНІ ТЕОРЕТИЧНІ ПІДХОДИ ДО ЗАХИСТУ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНОГО СЕРЕДОВИЩА

Можливі два підходи до побудови периметрів забезпечення ІБ ІКС – на основі рівнів або вимог безпеки і на основі загроз ІБ. У першому випадку периметри ІБ визначаються як умовні кордони, що розділяють зони з різними (необхідними) рівнями безпеки. На практиці периметри утворюються шляхом виділення деяких функціональних областей ІС з ідентичними вимогами забезпечення ІБ. У другому випадку периметри утворюються на основі можливих загроз ІБ.

Для ІКС пропонуються наступні периметри, див. рис. 1.1.

Рис. 1.1. Периметри забезпеченні ІБ ІКС

Стосовно до двох варіантів концептуального побудови ІКС пропонується наступний розподіл сервісів ІБ по периметрах ОБІ, див. рис. 1.2, 1.3.

Для побудови ефективної СЗІ, вибору і впровадженню адекватних технічних засобів повинен передувати опис, аналіз і моделювання загроз й уразливостей інформаційної системи та проведений на їх основі розрахунок й аналіз ризиків ІБ. Отже, очевидним є те, що спочатку кожна загроза повинна бути впізнана й ідентифікована.

Зазначимо, що використовувані в сучасних системах виявлення й протидії атакам (комп'ютерним нападам на інформацію – КНІ) (СВАП), які є невід'ємною складовою інформаційно-керуючих систем на у ЄС та США, методи є досить ефективними в тому разі, якщо відомі точні характеристики нападу на інформацію або загрози ІБ.

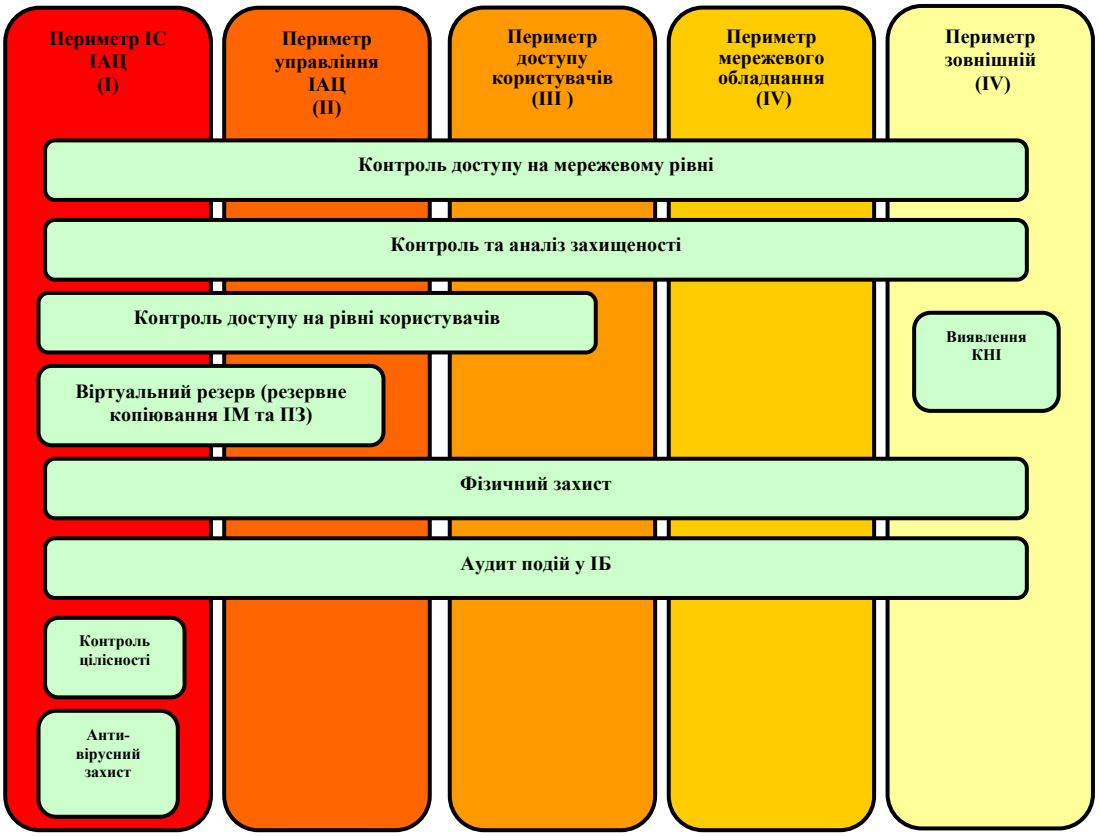


Рис. 1.2. Підсистеми ІБ для централізованого варіанта ІКС

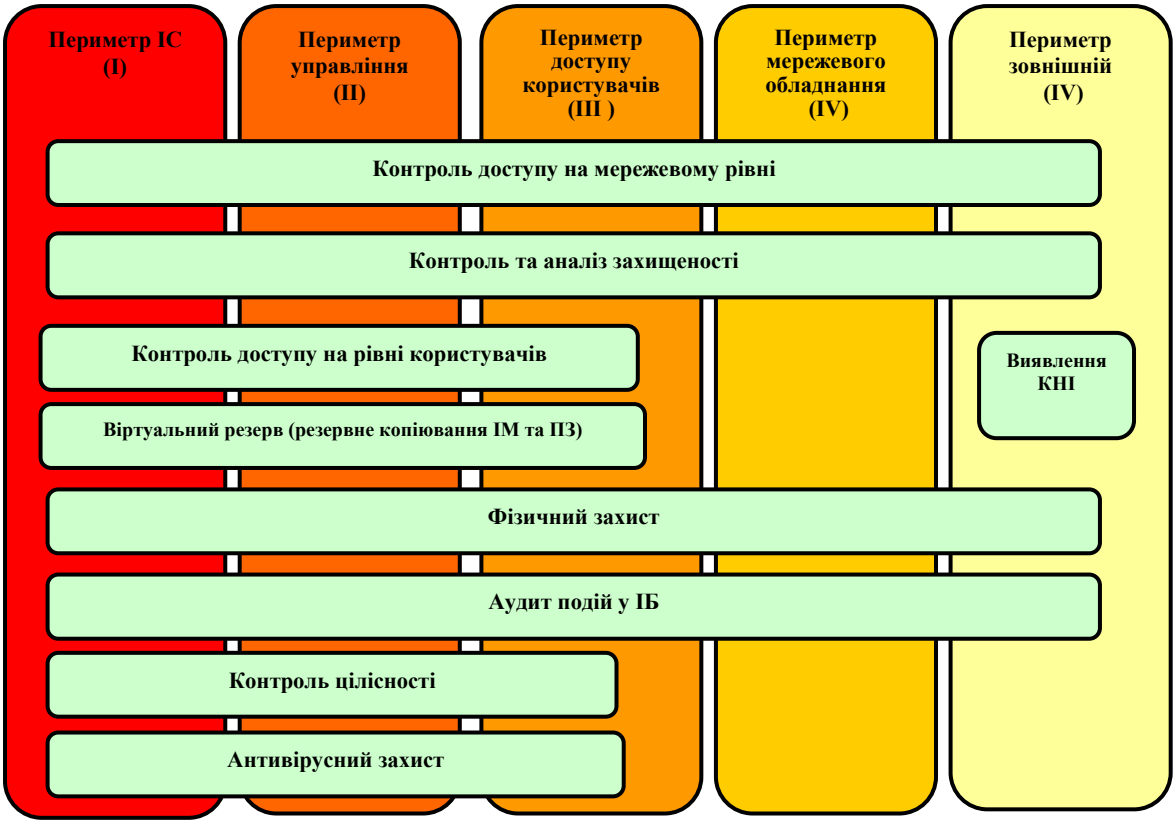


Рис. 1.3. Підсистеми ІБ для децентралізованого варіанта ІКС

Однак напади на інформацію у АС і мережні атаки, зокрема, постійно змінюються, наприклад, з появою нових технологій GSM-R, GPRS, GPS, LTE, Wi-Fi тощо, оскільки зловмисники використовують індивідуальні підходи, а також у зв'язку з регулярними змінами в ПЗ й апаратних засобах ІКС або АСК, див. табл. 1.4.

Таблиця 1.1

Сучасні системи виявлення й протидії КНІ

Метод (Модель)	Рівень спостереження	Аномалії/ Зловживання	Адаптивність	Обчислювальна складність	Використовується у СВВ	Контрольоване навчання	Розпізнавання Комбіно-ваних загроз ІКС
Модель переходів	Hybrid	-/+	-	o(N)	DREM, JANUS	-	-/+
Графи атак	Hybrid	-/+	+	NP	Bro	-	-/+
Нейронні мережі	NIDS	+/+	+	o(N) та вище	Hyperview	+	-/+
Імунні мережі	NIDS	+/+	+	o(N) та вище	NSL-KDD IDS	-/+	-
Експертні системи	NIDS	-/+	+	NP	NIDES, DIDS	-/+	-/+
Статистичні методи	NIDS	+/-	+	o(N)	NSM, Haystask	-/+	-/+
Кластерний аналіз	Hybrid	+/+	+	o(N) та вище	FIRE, Y-means	-/+	-/+
Поведінкова біометрія	NIDS	+/-	+	o(N) та вище	MDS, KDS	-/+	-
Сигнатурні методи	Hybrid	+/-	+	o(N) та вище	Snort, Sentarus IPS	-/+	-
HIDS - спостереження на рівні ОС вузла мережі (або спостереження на рівні мережевої взаємодії об'єктів на вузлах мережі); **Hybrid** - комбінація спостерігачів різних рівнів							

Незалежно від використовуваних методів виявлення нападів на інформацію, СВАП ІКС та АСК зустрічаються з однаковою проблемою – постійно змінювані характеристики нападів вимагають гнучкої СЗІ, яка здатна залишатися ефективною, навіть якщо не відомі точні характеристики нападу на інформацію, а також її ознаки [1-7 та ін.].

Неповнота інформації про загрози ІБ у ІКС та АСК є двоякою. По-перше, це часткова відсутність апріорної інформації, навіть на рівні уявлення про структуру всього об'єкта нападу на інформацію, що має, як правило, стохастичну природу. По-друге, обмежена можливість спостереження об'єкта

нападу й розпізнавання загроз, що належать певному класу. У граничному випадку заздалегідь відома лише загальна множина загроз ІБ і способів їх реалізації, див. рис. 1.4.

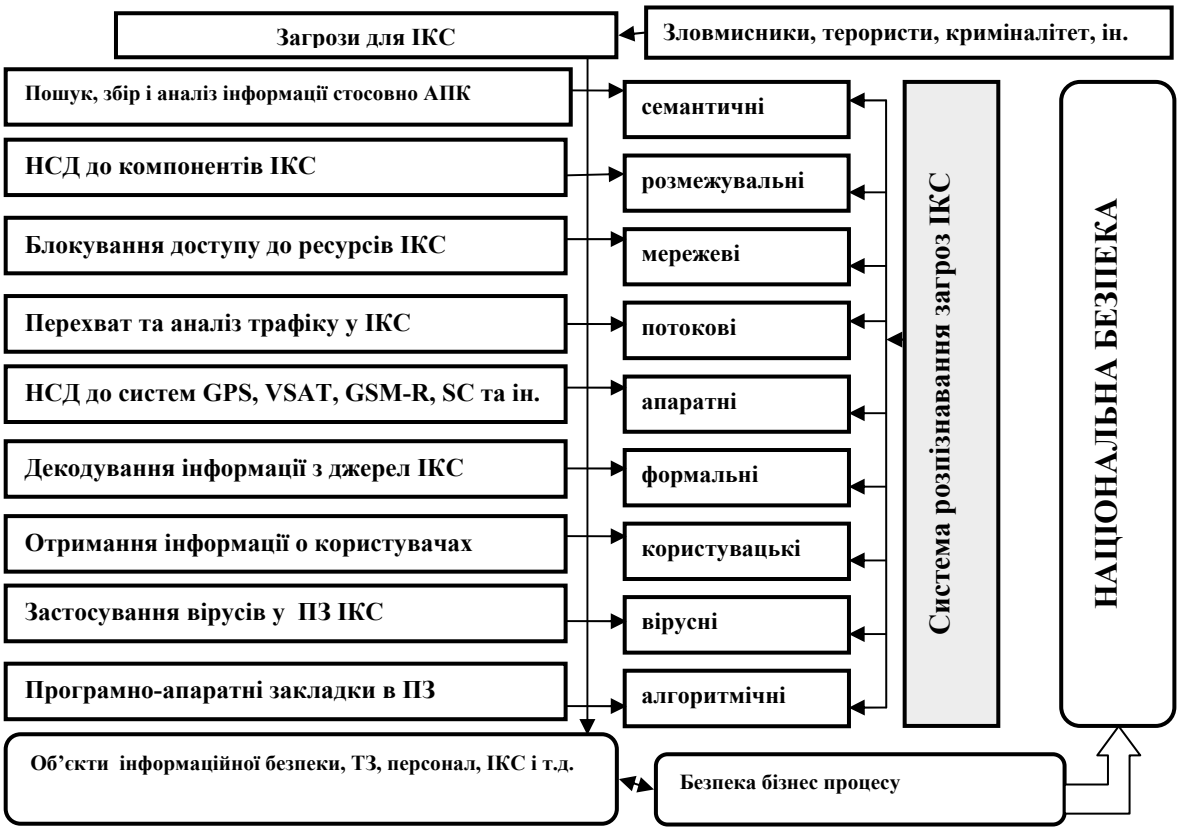


Рис. 1.4. Загрози для ІКС

Однак, як показує практика, одна з основних характеристик сучасних загроз полягає у тому, що вони довгий час не активуються, іноді до двох-трьох років [2, 4]. Цільові атаки, зокрема спрямовані на ІС підприємств, об'єкти інфраструктури, енергетики, транспорту тощо, зазвичай розробляються з урахуванням того середовища, на яке вони будуть націлені. Сучасні загрози створюються таким чином, щоб обійти захист, і, як правило, вже не виявляються за допомогою сигнатур. Розробка сценаріїв КНІ виконується з дотриманням всіх стандартів і технологій, з технічним завданням, робочим проектом, тестуванням, підтримкою і оновленням.

Вибір програмно-апаратного забезпечення захисту й проектування СЗІ ґрунтується на результатах такого аналізу з урахуванням економічної оцінки

співвідношення «вартість контрзаходів зі зниження ризиків/можливі втрати компанії від інцидентів ІБ». Концептуальна схема побудови ефективної системи безпеки ІС та АСК показана на рис. 1.5.

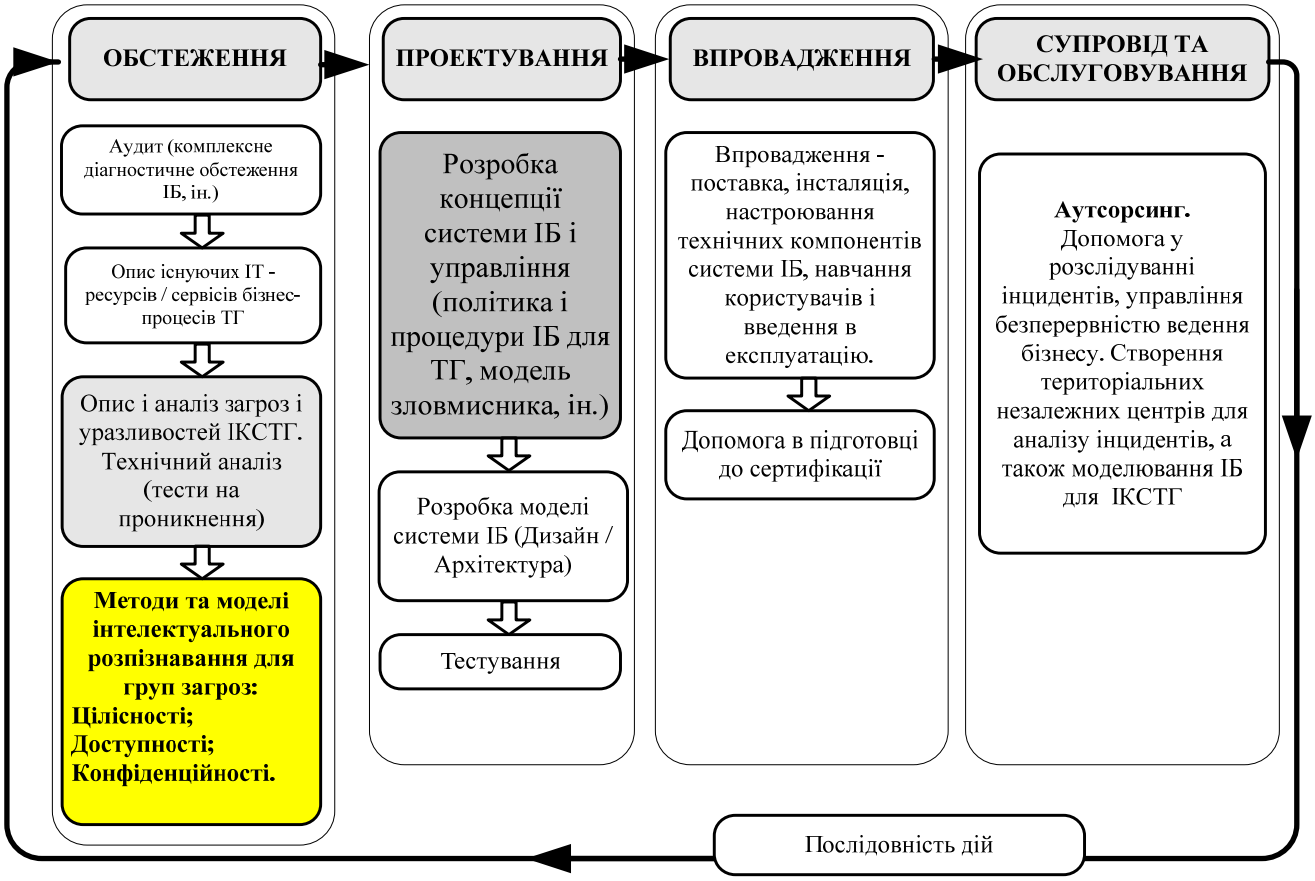


Рис. 1.5. Концептуальна схема побудови ефективної системи безпеки ІКС

Таким чином, політика захисту інформації та забезпечення ІБ ІКС на основі інтелектуального розпізнавання загроз (РЗ) повинна ґрунтуватись на комплексності і неперервності, де ключовим є врахування саме специфічних особливостей функціонування ІС, ВСК, КІС. Інформаційна інфраструктура багатьох галузей (транспорт, банки, промисловість, енергетика, АПК), та їх інформаційно-комунікаційні системи належить до найбільш уразливих об’єктів народного господарства.

Це дає змогу сформулювати наступні напрямки захисту ІКС: моніторинг обстановки та РЗ ІКС, зокрема у та ІКС, які визначені критичними для забезпечення проектного функціонування об’єктів, які захищаються; захист

каналів телекомунікацій (відео, радіо та ін. каналів зв'язку, навігації, комп'ютерних та Інтернет-мереж; контроль доступу до об'єктів захисту АСК; протидія спробам НСД зловмисників на захищені об'єкти ІКС, наприклад у нотації мови UML, що подана на рис. 1.6.

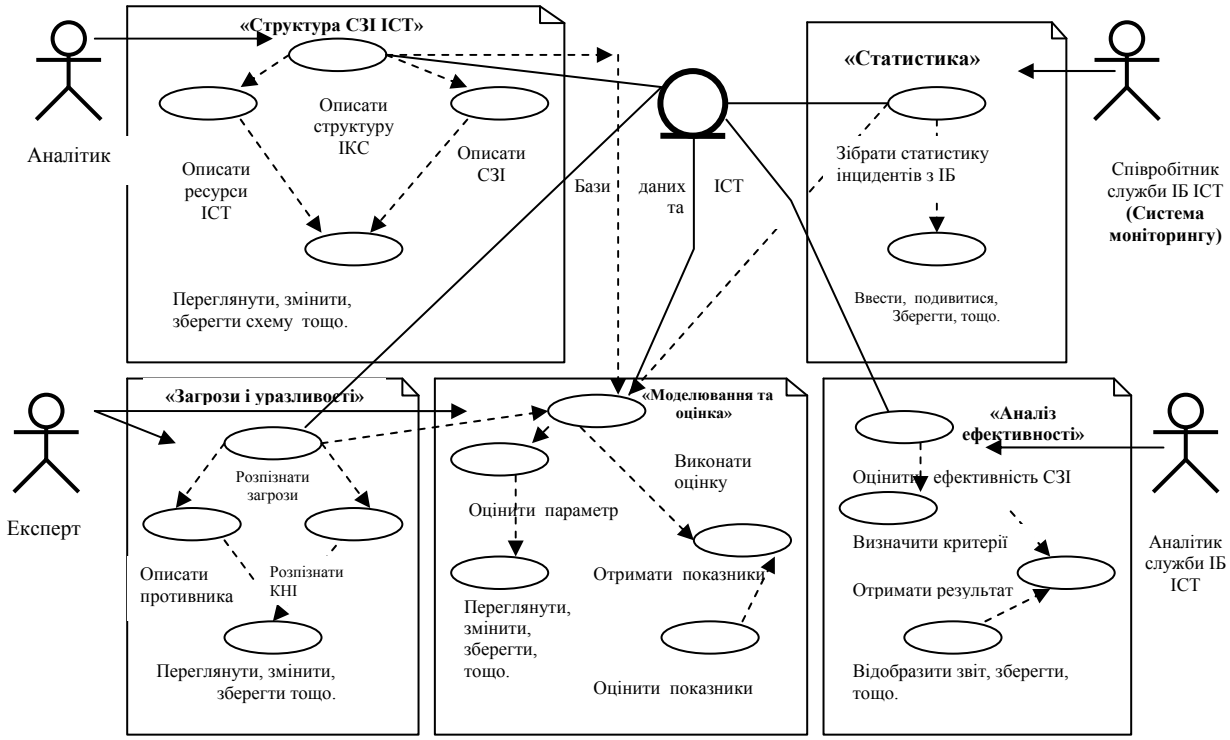


Рис. 1.6. Модулі СЗІ ІКС

Сьогодні існує багато робіт, що розкривають різні підходи до моделювання загроз ІБ ІС та АС, зокрема, КНІ [1-15]: мережі Петрі, метод аналізу зміни станів, емуляція нападів на інформацію у послідовному й паралельному режимах, причиново-наслідкова модель, концептуальні моделі КНІ, описові моделі мережі й зловмисників, структурований опис на базі дерев, моделювання "виживання" комп'ютерних систем, об'єктно-орієнтоване дискретне моделювання, модель запит/відповідь, ситуаційне вирахування й цілеспрямований виклик процедур, використання графів атак для аналізу уразливостей і т.д.

Варто зазначити, що підходи до створення СЗІ ІКС, які реалізують класичні алгоритми [1-15], підходять для строго певної кількості загроз. У тому випадку якщо виникає нова уразливість, і для неї здійснена загроза, яка не блокується СЗІ, необхідно передбачити можливість зміни архітектури базової системи й підсистему виявлення нових уразливостей і загроз. Зазначимо, що на сьогодні системи моделювання уразливостей і загроз не враховують ряд ризиків, і так само не враховують можливість появи нових уразливостей і загроз.

Тобто,

$$YZ_{1D_{сзі}} + YZ_{2D_{сзі}} = YZ_{3D_{сзі}} \neq YZ_{12D_{сзі}},$$

де $YZ_{iD_{сзі}}$ – уразливість рівнів СЗІ ІКС, АС.

Враховуючи надзвичайно велику різноманітність видів можливих атак на ІКС і критичність можливих наслідків їх реалізація для ІБ, необхідна розробка і впровадження в ІКС та АСК багаторівневої системи інтелектуального розпізнавання загроз (СІРЗ), захисту інформації та забезпечення ІБ.

РОЗДІЛ 2
МЕТОД ІНТЕЛЕКТУАЛЬНОГО РОЗПІЗНАВАННЯ ЗАГРОЗ БЕЗПЕЦІ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНОГО СЕРЕДОВИЩА

Винятково важливим є питання пріоритету при виборі конкретного набору актуальних загроз для ІКС. Таким пріоритетом у загальному випадку є вага, або ваговій коефіцієнт загрози, вимірюваний ймовірністю її реалізації. Саме ймовірності реалізації загроз є найбільш рухомою складовою проблеми формування політики ІБ. Сама по собі загроза не несе ніякої небезпеки, вона є лише припущенням про можливу небезпеку. Відповідно, у керівників підприємств АПК принаймні є два завдання. Перше завдання полягає в оцінці можливості реалізації передбачуваної загрози, а друге – в оцінці можливих витрат, що виникають при застосуванні відповідних засобів захисту.

2.1. Модель загроз інформаційно-комунікаційному середовищу

Оцінка можливості реалізації загрози залежить від багатьох факторів. У багатьох джерелах можливість реалізації загрози визначається як певна ймовірність [3, 6 та ін.]. Однак, загроза, яка є джерелом потенційного збитку, а тому представляє деяку небезпеку, будь-яким чином повинна бути виміряна. Фактично, мова йде про формування моделі загроз для ІКС, див. рис. 2.1. Безумовно, вимірювання загрози слід почати з оцінки можливості її виникнення. Така оцінка може бути зроблена на основі даних по відомим фактам появи загрози, вираженим через статистичну частотність. Дану оцінку можна розглядати як оцінку, засновану на наявному досвіді експлуатації СЗІ ІКС.

Другим фактором оцінки можливості реалізації загрози є оцінка витрат, що неминуче виникають при введенні тих чи інших засобів захисту.

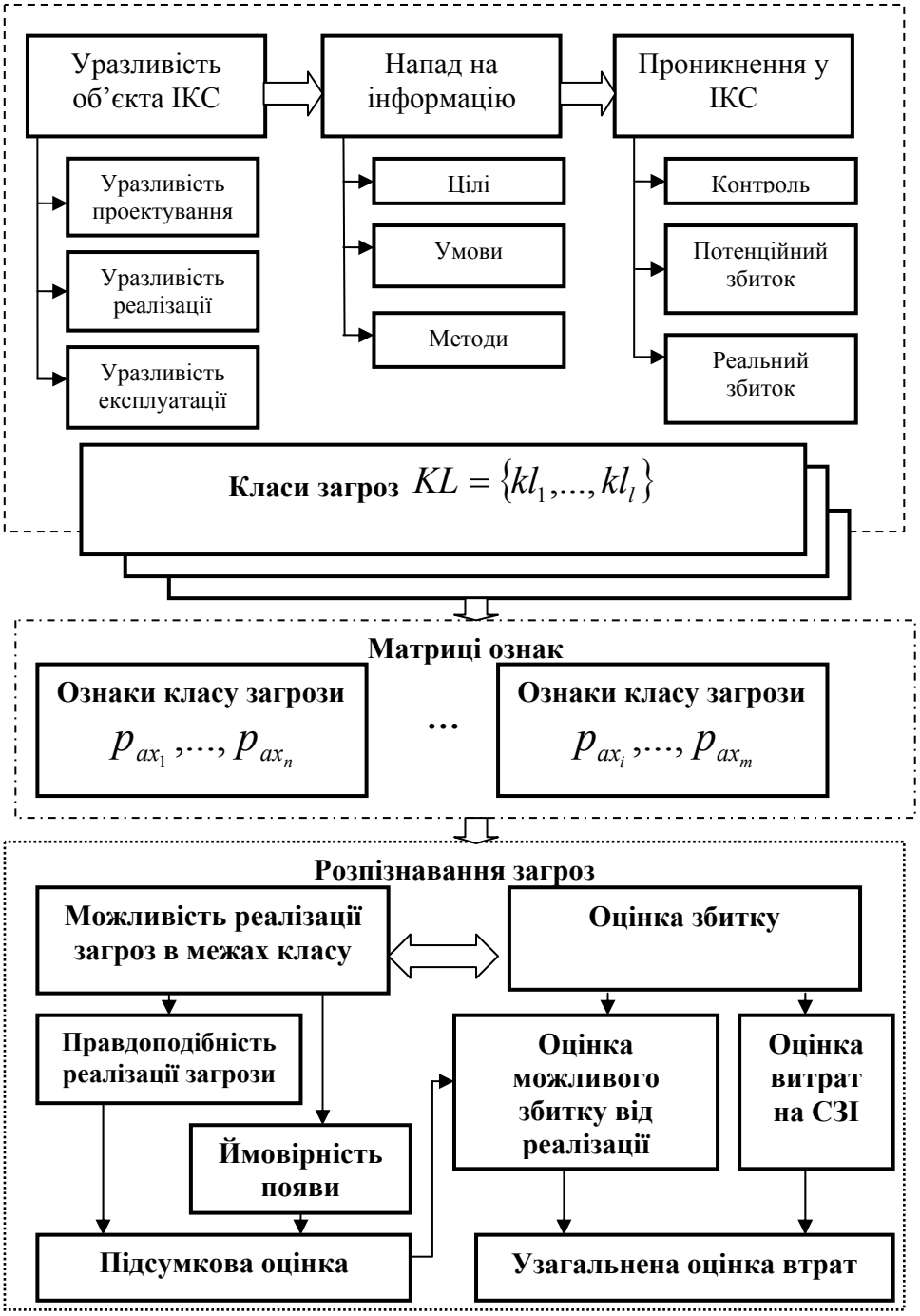


Рис. 2.1. Взаємозв'язок факторів категорії загроз ІКС

В силу того, що системи інтелектуального розпізнавання загроз для ІКС ще підлягають своєї реалізації, формалізована постановка задачі для їх розробки може бути сформульована таким чином.

Вихідними даними для всіх ІС є дані, що містяться в репозиторії *REP*:

$$REP = \langle SYS, Events, TAI, NIS, gov \rangle, \tag{2.1}$$

де SYS – дані про інфраструктуру ІКС, АСК та ін.), яке підлягає захисту (топологія, склад елементів, користувачі та ін.);

$Events$ – дані про події ІБ, які пройшли попередню обробку і знаходяться в репозиторії на зберіганні;

TAI – дані про сценарії атак (нападів на інформацію) у вигляді шаблонів;

NIS – дані про інциденти з ІБ, можливі контрзаходи і т. п.;

gov – вирішальне (розв'язувальне) в рамках політики безпеки.

Завдання, які вирішуються СІРЗ можуть бути записані таким чином:

Аналіз захищеності:

$$IOFP_j = FS(SYS, TAI, AT, gov), \tag{2.2}$$

де $IOFP_j$ – значення *j -го* показника захищеності;

AT – події ІБ, що відображають напад на інформацію (наприклад, атаку на ІКС, АСК та ін.);

FS – функція яка визначає $IOFP_j$ на основі прийнятої ПБ.

Управління кореляцією СІРЗ:

$$K_{\text{event}} = F\text{Cor}\{\text{e}_\text{i}\}, \tag{2.3}$$

де K_{event} – критична подія ІБ;

$e_i \subset Events$;

FCor – функція кореляції, яка дозволяє на основі аналізу подій з ІБ (зберігаються в репозиторії REP), виявляти критичні події.

Моделювання атак (нападів на інформацію):

$$ESC_{cr} = Model(SYS, TAI, AT, gov, T)\ , \tag{2.4}$$

де $ESC_{cr} \subset SYS$ – критичний елемент системи;

$Model$ – модель атаки у часі - T .

Підтримка прийняття рішень (або експертна система):

$$CM = \arg \quad \min \left| IOFP - IOFP_{\text{requirement}} \right|, \tag{2.5}$$

де $CM \subset gov$ – оптимальний контрзахід (СЗІ), що є елементом вирішального правила в рамках ПБ ІС АПК;

$IOFP \quad ma \quad IOFP_{\text{requirement}}$ – поточне та еталонне значення показника захищеності, відповідно.

Таким чином, формальна постановка задачі підтримки прийняття рішення (CM), є завданням синтезу. Цим вона принципово відрізняється від усіх попередніх, які є завданнями аналізу.

Візуальне представлення даних у випадку розпізнавання загрози ІБ

$$CM_v = FVizual\ (VF\ (SYS, Events, gov)), \tag{2.6}$$

де VF – візуальний інтерфейс аналітика СЗИ;

$FVizual$ – функція візуалізації, яка дозволяє приймати контрміру – CM_v .

Об'єктом атаки може стати будь-який з елементів ІКС, наприклад на підприємствах АПК, як елементі критично важливої інфраструктури держави. Проте в цілому всі елементи ІКС можуть бути віднесені до однієї з трьох категорій, див. рис. 2.2: центри опрацювання даних (ЦОД), АСК, АІС, ІС; периферійне обладнання та PLC; системи та канали зв'язку для обміну даними.

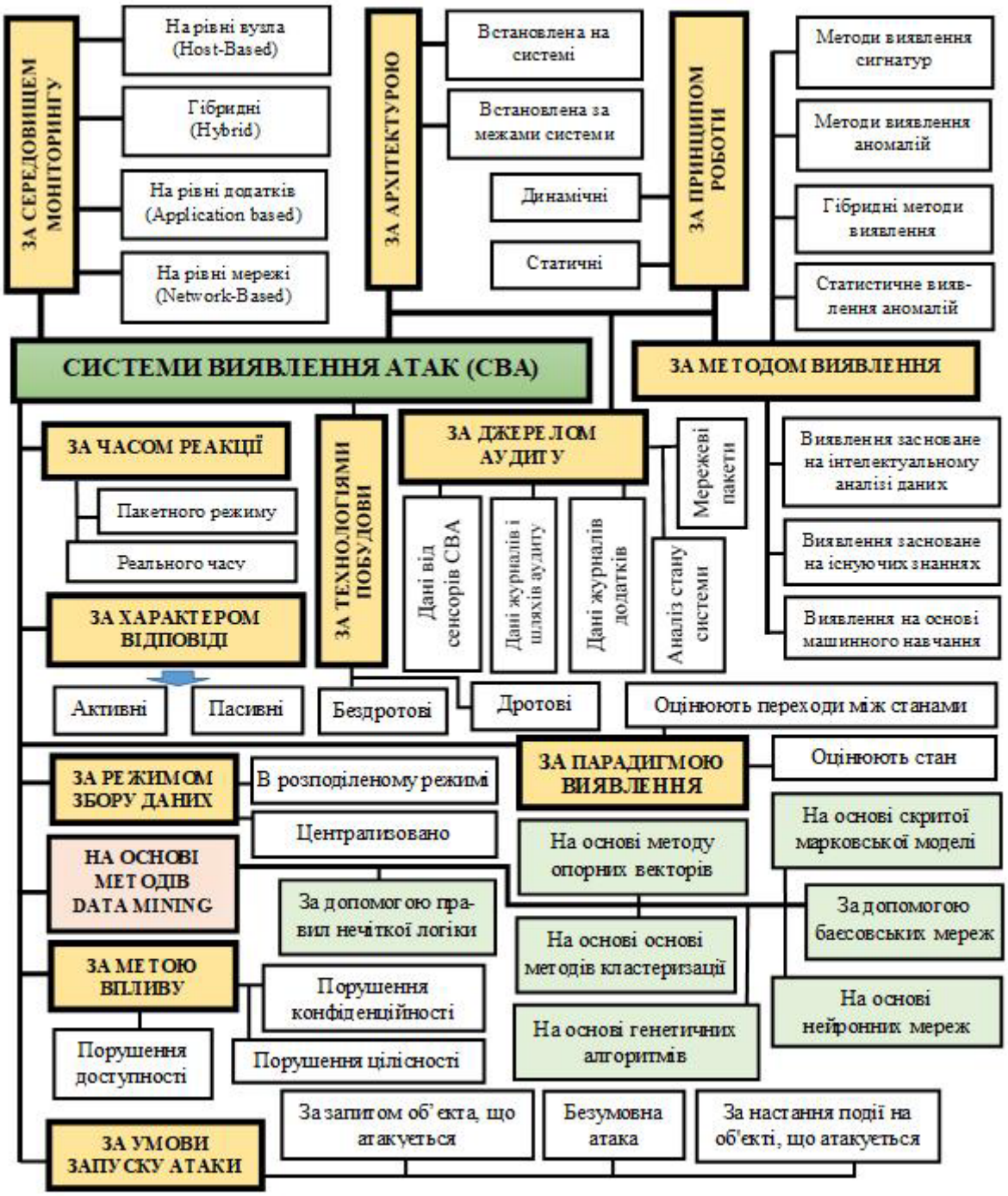


Рис. 2.2. Категорії елементів ІКС як об’єкти нападу на інформацію

Для ІКС можна виділити наступні основні види загроз, див. див. рис. 1.8, 2.2 та табл. 1.3: НСД до управління активним обладнанням та периферійними пристроями; НСД до даних, з метою їх підміни або знищення; комп’ютерний напад на інформацію ІКС з метою виведення систем з ладу в цілому або її окремих компонент.

Загроза зміни стану ІБ ІКС представлена у наступному вигляді:

$$S_R = \left\langle EUM^*, SDN, RDN, ADN, MIF, IR \right\rangle, \quad (2.7)$$

де EUM^* – множина сутностей, до складу якої входить: підмножина вузлів ІКС - um^* (потенційні уразливості);

SDN – множина суб'єктів ІКС;

RDN – множина ребер графа станів системи S_R, у тому числі тих, що відповідають правам доступу користувачів до EUM^*;

ADN – множина ребер графа станів системи S_R, що відповідають отриманому доступу до EUM^*;

MIF – множина ребер графа станів системи S_R, що відповідають інформаційним потокам між EUM^* ($um^* \subset EUM^*$);

IR – функція ієрархії EUM^*.

Традиційні засоби захисту в архітектурі системи інтелектуального розпізнавання загроз займають нижній рівень і відповідають, крім реалізації функцій апріорного захисту, за формування та надання даних про події ІБ. На другому рівні СІРЗ здійснюється ведення репозиторію даних про ІБ. На вищому рівні СІРЗ виконується управління кореляцією, аналіз захищеності ІКС, моделювання атак, підтримка прийняття рішень з забезпечення захисту інформації та ІБ, а також візуальний аналізу даних.

2.2. Метод і модель інтелектуального розпізнавання загроз інформаційно-комунікаційному середовищу, засновані на побудові покриттів класів

Сучасні інформаційні та інформаційно-керуючи системи, наприклад, Gonrand, Videotrans, ISCIS, BRS та ін., оснащені багаторівневими засобами

виявлення нападу на інформацію і запобігання НСД (так звані системи виявлення та протидії комп'ютерним атакам – СВАП) [1-15].

Політика безпеки в межах СВАП поширюється на всі елементи інформаційної мережі підприємства – комутатори і маршрутизатори, ПК, стільникові корпоративні та IP-телефони, бездротові точки доступу і т. д. [1].

Серед методів, використовуваних у СВАП, можна виділити два напрямки: одні спрямовані на виявлення аномалій у системі, що захищається, а інші – на пошук зловживань [11-15].

Кожен зі згаданих напрямків має свої переваги та недоліки, тому в більшості існуючих СВАП застосовуються комбіновані рішення, засновані на синтезі відповідних методів. Ідея методів, які використовуються для виявлення аномалій, полягає в тому, щоб розпізнати, чи є процес, що викликав зміни в роботі системи, діями зловмисника. Виділяють дві групи методів: з контрольованим навчанням і з неконтрольованим навчанням. Основна відмінність між ними полягає в тому, що методи контрольованого навчання використовують фіксований набір параметрів оцінки (ознак) і певні апріорні відомості про значення ознаках, при цьому час навчання фіксований. У неконтрольованому навчанні множина ознак може змінюватися із часом, зі збільшенням кількості загроз і уразливостей, а процес навчання відбувається постійно. Мета другого напрямку (виявлення зловживань) – пошук послідовностей подій, що визначені (адміністратором з безпеки або експертом під час навчання СВАП) як етапи реалізації нападу на інформацію [1-15].

На сьогодні виділяють лише методи з контрольованим навчанням. Ті, що реалізовані на даний момент у СВАП, засновані на загальних представленнях теорії розпізнавання образів. Згідно з ними, для виявлення аномалії на підставі експертної оцінки формується образ нормального функціонування ІС. Цей образ виступає як сукупність значень параметрів оцінки, тобто ознак. Його зміна вважається проявом аномального функціонування системи. Після виявлення аномалії та оцінки її ступеня формується судження про природу змін: вони є наслідком нападу на інформацію або

припустимим відхиленням. Для виявлення зловживань також використовується образ (сигнатура), однак тут він відображає заздалегідь відомі дії зловмисника [1-15].

Методи виявлення аномалій спрямовані на виявлення невідомих комп'ютерних нападів на інформацію. Для захищуваної ІС (АС або АСК) на основі сукупності параметрів оцінки формується «образ» нормального функціонування. У сучасних СВАП виділяють кілька способів побудови «образу»: нагромадження найбільш характерної статистичної інформації для кожного параметра оцінки; навчання нейронних мереж значеннями параметрів оцінки; подійне представлення [3].

Легко помітити, що у виявленні значну роль відіграє множина ознак, тобто параметрів оцінки. Тому у виявленні аномалій одним із головних завдань є вибір оптимальної кількості ознак.

Не менш важливим завданням є також визначення загального показника аномальності. Складність полягає в тому, що ця величина повинна характеризувати загальний стан «аномальності» захищуваної ІКС.

У наш час використовується евристичне визначення множини параметрів вимірів захищуваної ІКС. Використання даної множини повинне дати найбільш ефективне й точне інтелектуальне розпізнавання нападів на інформацію [1-15]. Складність вибору множини можна пояснити тим, що складові підмножини залежать від типів виявлених загроз, див. рис. 2.3. Тому та сама сукупність параметрів не буде адекватною для всіх типів нападів на інформацію.

Кожну ІКС, що складається зі звичних апаратних і програмних засобів, можна розглядати як унікальний комплекс зі своїми особливостями. Це пояснює можливість пропуску СВАП специфічних для захищуваних ІКС нападів на інформацію у ситуаціях, коли застосовується той самий набір параметрів оцінки загроз ІБ. Найкраще рішення - визначення необхідних параметрів оцінки в процесі роботи. Труднощі ефективного динамічного формування параметрів оцінки загрози ІБ полягають у тому, що розмір зони

пошуку експоненційно залежить від потужності початкової множини. У зв'язку з наявністю значної кількості факторів різної природи, функціонування ІКС і СВАП має ймовірнісний характер. Особливо потрібно виділити завдання обґрунтування функції втрат ІКС, що задається відповідно до її цільової функції на зоні параметрів функціонування системи. При цьому цільова функція повинна бути визначена не тільки на експертному рівні, але й відповідно до сукупності параметрів функціонування всієї інформаційної системи й завдань, що покладаються на неї. Тоді показник якості СВАП буде визначатися як один із параметрів, що впливають на цільову функцію, а його припустимі значення стануть припустимими значеннями функції втрат.

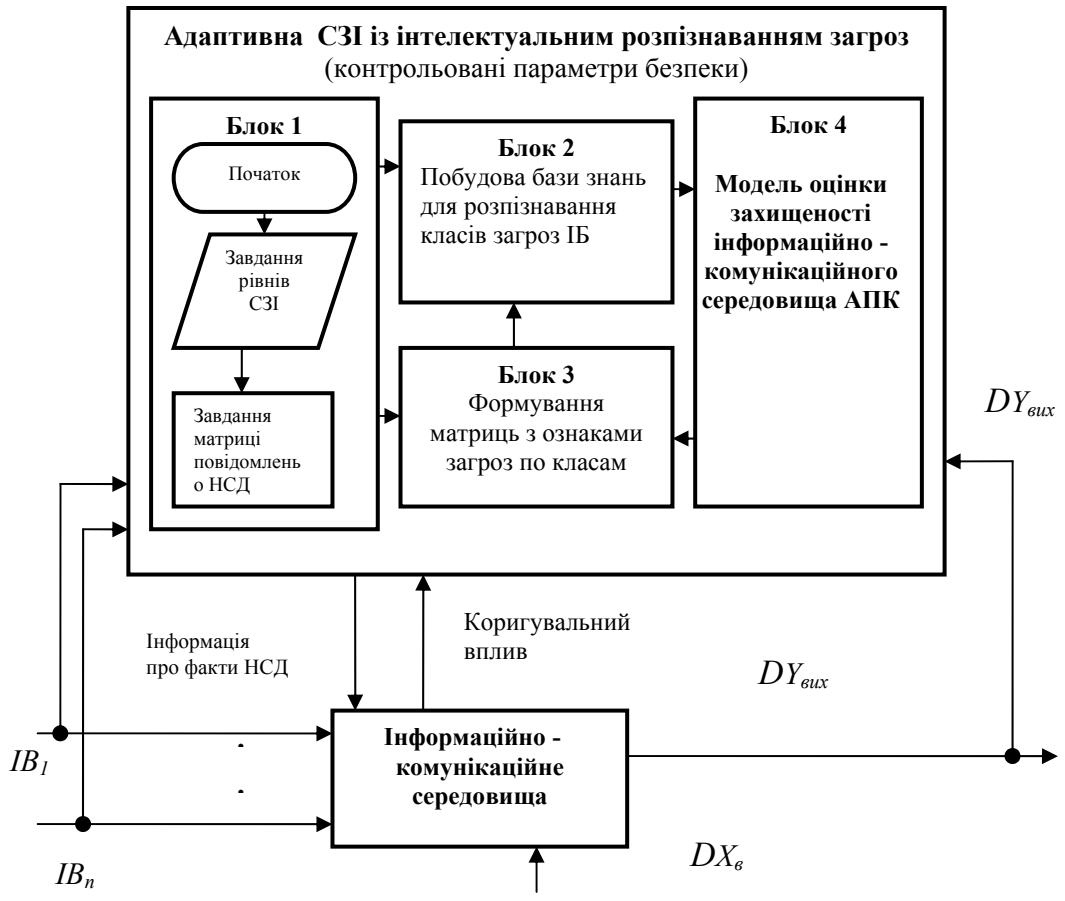


Рис. 2.3. Схема адаптивної системи інтелектуального розпізнавання загроз та захисту ІКС

Після обґрунтування законів і функцій реальним завданням є одержання формалізованими методами оптимальної структури СВАП ІКС у вигляді

сукупності математичних операцій. Таким чином може бути вирішене завдання синтезу структури СВАП для конкретної ІКС. На підставі отриманих математичних операцій можна буде розрахувати залежності показників якості функціонування СВАП від параметрів її функціонування, а також від параметрів функціонування інформаційної системи, тобто стане можливим реальний аналіз якості функціонування СВАП.

Складність застосування до СВАП формалізованого апарату аналізу й синтезу ІКС полягає в тому, що конкретний інформаційний комплекс і його підсистема ІБ – СВАП складаються з різнорідних елементів, які можуть описуватися різними розділами теорії (системами масового обслуговування, кінцевими автоматами, теорією ймовірностей, теорією розпізнавання образів тощо), тобто розглянутий об'єкт дослідження є агрегативним. Тому математичні моделі можна одержати тільки для окремих складових частин ІКС, що ускладнює аналіз і синтез СВАП у цілому, але подальша конкретизація застосування формалізованого апарату аналізу й синтезу дозволить оптимізувати СВАП.

Основними принципами створення СВАП як елемента комплексів СЗІ є такі [1-15]:

1) принцип прозорості. СВАП повинна функціонувати у фоновому режимі непомітно для користувачів ІКС, не знижуючи оперативності виконання технологічних циклів керування бізнес-процесами, при цьому забезпечуючи виконання своїх цільових функцій;

2) принцип оптимальності. Розробка СВАП повинна проводитися з урахуванням того, що кожен із методів виявлення (протидії) дозволяє досить ефективно й вірогідно виявляти (нейтралізовувати) тільки певні види нападів на інформацію;

3) принцип адекватності. Розроблювані для реалізації в СВАП проектні рішення повинні бути диференційовані залежно від частоти, ймовірності й очікуваного збитку від успішної реалізації кожного виду нападу на інформацію;

4) принцип повноти. Даний принцип полягає у використанні для

виявлення нападу на інформацію, стану і значення основних параметрів усіх програмних і технічних елементів пунктів керування ІКС;

5) принцип адаптивності. СВАП повинні створюватися з урахуванням того, що з розвитком ІКС буде відбуватися поступова зміна складу й характеристик програмних і технічних засобів АС, що, у свою чергу, призведе до розширення переліку загроз безпеки. Тому при створенні СВАП у її складі повинні бути передбачені механізми адаптації ІКС до мінливих умов функціонування.

Використання розглянутих принципів, дозволить створювати ІКС, що забезпечуватимуть ефективне й своєчасне виявлення та протидію спробам реалізації КНІ при збереженні оперативності виконання технологічних циклів керування.

Звичайний недолік більшості СВАП – помилкові спрацьовування, оскільки в них майже завжди задіяна тільки одна технологія виявлення (зазвичай, ідентифікація КНІ). При аналізі використовуються сигнатури, набір правил політики безпеки, установки протоколу та статистичні дані аномальної роботи мережі. Однак при виникненні нових загроз і уразливостей для інформаційних ресурсів ІКС подібний підхід не завжди може сприяти ефективному захисту інформації.

Тому нижче зупинимося на запропонованому методі і моделі інтелектуального розпізнавання загроз для безпеки ІКС, заснованих на побудові покриттів класів загроз ІБ.

Нехай існує ряд загроз об'єкту інформаційної безпеки (ОІБ) (загальна класифікація загроз наведена на рис. 1.20). Ступінь небезпеки кожної загрози залежить від значень ряду факторів, що підвищують або знижують захищеність ОІБ від даної загрози. Фактори, що знижують захищеність ОІБ, будемо називати факторами ризику, а ті, що підвищують її – факторами захищеності. Інтегральна оцінка загроз й захищеності ОІБ є функцією його захищеності від кожного виду загроз. Варто зазначити, що питання класифікації загроз інформаційної безпеки дотепер активно досліджуються [1-15]. Закріпленого

законодавчо виду класифікації загроз не існує, а тому кожний розробник АС або аудитор будує зручну для себе класифікацію загроз інформаційної безпеки. Основне ж завдання розробки класифікації загроз полягає в тому, щоб забезпечити зручність її використання на практиці та забезпечити можливість пов'язати цю класифікацію з подіями безпечного функціонування ІКС.

Існує кілька варіантів формалізації залежності ступені захищеності ОІБ від факторів:

1) кожна загроза ІКС залежить від одного фактору, тобто між ступенем загрози й значеннями фактору існують відносини взаємооднозначної відповідності;

2) кожна загроза ІКС залежить від значень багатьох факторів;

3) ті самі фактори в загальному випадку впливають на ступінь захищеності ОІБ не від одного, а від багатьох видів загроз.

Для забезпечення однозначності, повноти й цілісності класифікації уведемо такі вимоги до класифікації:

1) непересічні класи (визначає однозначність вибору класу на підставі зовнішнього вирішального правила);

2) застосовність (додавання класу не повинне викликати дроблення більше ніж одного класу на дві частини);

3) об'єктивність (наявність або відсутність класу повинна підтверджуватися відомими класифікаціями);

4) розширюваність (додавання класу можливе шляхом дроблення існуючих класів);

5) кількість класів є остаточною.

Як показав аналіз публікацій [1-15 та ін.], у наш час відбувається інтенсивний пошук нових методів аналізу даних у СВАП і все більше уваги приділяється застосуванню методів інтелектуального аналізу даних (ІАД, Data Mining).

Інформація, яка є основою побудови ДПРЗ ІБ, може бути подана в різних формах, наприклад, у вигляді важко з'ясовних ознак КНІ або НСД

$\{p_{ax1},...,p_{axn}\}$ у комп'ютерних системах, діапазонів граничних значень, параметрів вхідного вихідного трафіка, непередбачуваних адрес пакетів, атрибутів, часових параметрів, запитів і т. д.

Позначимо через *MI* загальне число загроз ІБ; *PA* – число можливих цілей порушника в захищеному ІКС; B_{p_a} – множина номерів загроз інформації, реалізованих порушником при досягненні p_a *-ї* мети.

Стандартна постановка завдання розпізнавання, у нашому випадку мова йде про розпізнавання нападу на інформацію або спроби НСД до ІКС, полягає в такому [1-15]. Досліджується деяка множина об'єктів, у нашому випадку це *PA* – число можливих цілей порушника. Об'єкти цієї множини описуються множиною ознак $\{p_{ax1},...,p_{axn}\}$ [1-15]. Відомо, що множина *PA* представлена у вигляді об'єднання непересічних підмножин (класів) загроз інформації – *KL*. Існує остаточний набір об'єктів $\{sp_{a1},...,sp_{am}\}$ з *PA*, про які відомо, до яких класів вони належать (це прецеденти, тобто об'єкти, використовувані для навчання – ОВН). Потрібно за пред'явленим набором значень ознак, тобто описом деякого об'єкта sp_{an} з *PA*, про який невідомо, до якого класу він належить, визначити цей клас і, відповідно, вибудувати роботу СЗІ таким чином, щоб вона могла ефективно протидіяти загрозі в межах даного класу (розділ 1).

Для вирішення прикладних завдань розпізнавання, наприклад, для автоматизації пошуку нових загроз і уразливостей у ІКС, можна використовувати методи, засновані на комбінаторному аналізі ознакових описів об'єктів $\{sp_{a1},...,sp_{am}\}$, які особливо ефективні у випадку, коли інформація цілочислена й число припустимих значень кожної ознаки невелике. Основними роботами з цього питання є роботи Ю. І. Журавльова, С. В. Яблонського та М. Н. Вайнцвайга, С. Ханта [1, 2].

Класифікація атак представляє собою сукупність можливих варіантів дій джерела загроз певними методами реалізації із використанням уразливостей, які призводять до реалізації цілей атаки [7].

Ціль атаки може не збігатися з метою реалізації загроз і бути спрямованою на одержання проміжного результату, необхідного для подальшого досягнення реалізації загрози. У разі, якщо цілі атаки не збігаються з метою реалізації загрози, сама атака розглядається як етап підготовки до здійснення дій, спрямованих на реалізацію загрози, тобто як «підготовка до здійснення» НСД до ІКС.

Головною особливістю запропонованого методу інтелектуального розпізнавання загроз ІКС, є можливість одержання результату за відсутності інформації про функції розподілу значень ознак і за наявності малих навчальних вибірок. Також непотрібно задавати метрику в просторі описів об'єктів. У цьому разі для кожної ознаки визначається бінарна функція близькості між його значеннями, що дозволяє розрізняти об'єкти та їх підописи [1-15].

Основним завданням побудови ДПРЗ ІБ є пошук інформативних підописів (або фрагментів описів) об'єктів.

Інформативними будемо вважати такі фрагменти, які відображають певні закономірності в описах об'єктів, використовуваних для навчання, тобто наявність або, навпаки, відсутність цих фрагментів у класифікованому об'єкті дозволяє судити про його приналежність до того або іншого класу. У ДПРЗ ІБ інформативними вважаються такі фрагменти, які зустрічаються в описах об'єктів одного класу, але не зустрічаються в описах об'єктів інших класів. Розглянуті фрагменти, зазвичай, мають змістовний опис у термінах проектування СЗІ.

При побудові ДПРЗ ІБ для СЗІ ІКС вводиться поняття елементарного класифікатора (ЕК) [1]. Під ЕК розуміють фрагмент опису об'єкта, використовуваного для навчання. Для кожного класу KL будується деяка множина ЕК із заздалегідь заданими властивостями та, як правило, використовуються класифікатори, які зустрічаються в описах об'єктів одного класу й не зустрічаються в описах об'єктів інших класів, тобто характеризують лише деякі з ОВН даного класу. З іншого боку, набори значень ознак, що не зустрічаються в описі жодного з ОВН класу, характеризують усі об'єкти даного

класу, отже, є більш інформативними. Тому актуальним є питання конструювання ДПРЗ ІБ, заснованих на принципі «незустрічальності» наборів із припустимих значень ознак, при цьому завдання полягає в побудові такого розв'язувального (вирішального) правила $gov(p_{axi})$, при якому розпізнавання загроз ІБ проводилося б мінімальною кількістю помилок.

Інша проблема – наявність у вибірці ОВН таких об'єктів, які лежать на межі між класами *KL*. Кожний такий об'єкт не є «типовим» для свого класу, оскільки його опис схожий на описи об'єктів з інших класів. Наявність нетипових об'єктів збільшує довжину фрагментів, що розрізняють об'єкти з різних класів. Довгі фрагменти рідше зустрічаються в нових об'єктах, тим самим збільшується число нерозпізнаних об'єктів.

Необхідність побудови ефективних реалізацій для ДПРЗ ІБ прямо пов'язана з питаннями вивчення метричних (кількісних) властивостей множини інформативних фрагментів. Важливими й технічно дуже складними в СЗІ є завдання одержання асимптоматичних оцінок для типових значень числа (тупикових) покриттів і довжини (тупикового) покриття цілочисленої матриці, а також завдання одержання аналогічних оцінок для припустимих і максимальних кон'юнкцй логічної функції, використовуваних для синтезу схемотехнічних рішень апаратної частини СЗІ.

При вирішенні завдання, пов'язаного із проектуванням ефективної СЗІ ІКС, достовірна інформація про структуру множини *PA*, як правило, відсутня, тому при побудові алгоритму ДПРЗ ІБ ми не можемо гарантувати якість роботи цього алгоритму на нових (відмінних від $\{sp_{a1},...,sp_{am}\}$) об'єктах. Однак, якщо ОВН досить характерні для досліджуваного множини об'єктів, то алгоритм, що рідко помиляється при навчанні, буде давати непогані результати і на невідомих (що не входять до навчальної вибірки) об'єктах. У зв'язку із цим велику увагу приділено проблемі коректності розпізнавальних алгоритмів. Алгоритм є коректним, якщо всі об'єкти з навчальної вибірки він розпізнає правильно.

Найпростішим прикладом коректного алгоритму є такий. Розпізнаваний об'єкт sp_{an} порівнюється з кожним з ОВН $\{sp_{a1},...,sp_{am}\}$. У випадку якщо опис об'єкта sp_{an} збігається з описом ОВН об'єкта sp_{ai} об'єкт sp_{an} належить до того класу, до якого належить об'єкт sp_{ai}, а якщо ні, то алгоритм відмовляється від розпізнавання. Нескладно помітити, що описаний алгоритм є коректним, однак він не зможе розпізнати жоден об'єкт sp_{ai}, опис якого не збігається з описом жодного з ОВН.

Очевидно, що вимога повного збігу описів розпізнаваного об'єкта й одного з ОВН є занадто обережною. Аналіз різновидів КНІ і типів НСД до ресурсів ІКС свідчить про те, що питання про близькість об'єктів sp_{ai} та їх приналежність до одного класу можна вирішувати на підставі порівняння деякої множини їх підописів. Тому виникає проблема, як обирати піднабори ознак, що породжують такі підописи, за якими будуть порівнюватися об'єкти. Один із варіантів відповіді на дане питання використовується в моделі алгоритмів обчислення оцінок (АВО) [1-12].

Розглянемо завдання класифікації для двох класів загроз ІБ. Припустимо, що на практиці аналітик служби ІБ має у своєму розпорядженні вихідні дані, які характеризують нормальну активність користувачів ІКС, і деякі приклади нападу на інформацію. Будемо вважати, що ці класи не перетинаються. Очевидно, що існує одиничний вектор VE і число $ЧХ$, для яких слушними є такі відносини: $(p_{ax_1}, VE) > ЧХ$ при $p_{ax_1} \in P_{ax_1}$ й $(p_{ax_1}, VE) < ЧХ$ при $p_{ax_2} \in P_{ax_2}$, де P_{ax_1}, P_{ax_2} – простір спостережень, що сприймаються спостерігачем – простір ознак (тобто деякий кількісний вимір об'єкта). У такому разі говорять, що P_{ax_1} й P_{ax_2} розділені гіперплощиною. Завдання полягає у знаходженні оптимальної розділювальної гіперплощини, формально, відповідної вектору $VE_{опт}$, при якому досягається максимум (тобто розділювальна гіперплощина повинна бути розташована максимально далеко від найближчих до неї точок обох

класів). У роботі [3] доведена теорема, що якщо 2 множини P_{ax_1} й P_{ax_2} розділені гіперплощиною, то оптимальна розділювальна гіперплощина існує і єдина.

Однак на практиці вибірка рідко є лінійно роздільною. Тому потрібно здійснити перехід від вихідного простору ознакових описів об'єктів P_{ax_j} до нового простору NEP_{ax} за допомогою деякого перетворення $P_{ax_j} \to NEP_{ax}$. Якщо вибірка в P_{ax_j} не суперечлива й NEP_{ax} має досить високу розмірність, то завжди найдеться простір, у якім вона роздільна. Простір NEP_{ax} називають випростовувальним.

Уведемо наступні позначення. Нехай NP_{p_a} – деякий набір з r_{p_a}, $r_{p_a} \le MI$ різних цілочислених ознак виду $\{p_{aj_1},..., p_{aj_r}\}$. Близькість об'єктів $sp'_a = (\alpha p'_{a1}, \alpha p'_{a2},..., \alpha p'_{aMI})$ і $sp''_a = (\alpha p''_{a1}, \alpha p''_{a2},..., \alpha p''_{aMI})$ з PA за набором ознак NP_{p_a} будемо оцінювати величиною:

$$BN(sp'_a, sp''_a, NP_{pa}) = \begin{cases} 1, & якщо \quad \alpha \mathrm{p}'_{\mathrm{j_{ti}}} = \alpha \mathrm{p}''_{\mathrm{j_{ti}}} \quad при \quad \mathrm{ti} = 1,2,..., \mathrm{r_{p_a}}, \\ 0 & у \quad протилежному \quad випадку. \end{cases} \tag{2.8}$$

Таким чином, принципова схема побудови АВО для СЗІ буде такою. У множині ознак $\{p_{a_1},..., p_{a_{jMI}}\}$ виділяється сукупність різних підмножин виду $NP_{p_a} = \{p_{aj_1},..., p_{a_{MI}}\}$, $r_{p_a} \le MI$. Надалі виділені підмножини називаються опорними множинами алгоритму, а вся їхня сукупність позначається через ΩMI.

Далі задамо наступні параметри:

po_{sp_a} – параметр, що характеризує значущість мети (об'єкта) sp_{ai}, i=1, 2,..., PA;

$po_{NP_{pa}}$ – параметр, що характеризує значущість об'єкта опорної множини $NP_{p_a} \in \Omega MI$.

Далі проводиться процедура обчислення оцінок. Розпізнаваний об'єкт sp_{an} порівнюється з кожним ОВН sp_{ai} за кожною опорною множиною.

Для кожного класу загроз ІКС KL, $KL \in \{KL_1,...,KL_l\}$, обчислюється оцінка приналежності $\Gamma(sp_a, KL)$ об'єкта sp_a до класу KL, яка має вигляд:

$$\Gamma(sp_a, KL) = \frac{1}{|LW_{KL}|} \sum_{sp_{ai} \in KL} \sum_{NP_{pa} \in \Omega MI} po_{sp_a} \cdot po_{NP_{pa}} \cdot BN(sp_a, sp_{ai}, NP_{p_a}), \quad (2.9)$$

де $|LW_{KL}| = |KL \cap \{sp_{a1},...,sp_{aMI.}\}|$.

Об'єкт sp_{an} належить до того класу, який має найбільшу оцінку. Якщо класів з найбільшою оцінкою небагато, то відбувається відмова від розпізнавання. Очевидно, що побудований алгоритм не завжди є коректним. Для коректності цього алгоритму потрібне виконання системи лінійних нерівностей зазначеного нижче вигляду:

$$\Gamma(\mathrm{sp}_{\mathrm{a1}}, KL_1) > \Gamma(\mathrm{sp}_{\mathrm{a1}}, KL_2), \quad \Gamma(\mathrm{sp}_{\mathrm{aMI}_1}, KL_1) > \Gamma(\mathrm{sp}_{\mathrm{aMI}_1}, KL_2),$$
$$\Gamma(\mathrm{sp}_{\mathrm{aMI}_{1+1}}, KL_2) > \Gamma(\mathrm{sp}_{\mathrm{aMI}_{1+1}}, KL_1) \ldots \Gamma(\mathrm{sp}_{\mathrm{aMI}}, KL_2) > \Gamma(\mathrm{sp}_{\mathrm{aMI}}, KL_1).$$

Рішення системи зводиться до вибору параметрів $po_{sp_{ai}}$ i = 1, 2,..., PA, та $po_{NP_{pa}}$, $NP_{p_a} \in \Omega MI$. У разі, якщо система несумісна, знаходиться її максимальна спільна підсистема й з рішення цієї підсистеми визначаються значення параметрів $po_{sp_{ai}}$ і $po_{NP_{pa}}$.

Інший спосіб добитися коректності алгоритму – обрати «гарну» систему опорних множин. Зокрема, обрати її так, щоб для будь-якого ОВН $sp'_a \notin KL$ була виконана умова $\Gamma(sp'_a, KL) = 0$ й для будь-якого ОВН $sp''_a \in KL$ було виконано $\Gamma(sp''_a, KL) > 0$. Це можна зробити так.

Нехай $NP_{p_a} = \{p_{aj_1}, ..., p_{a_{MI}}\}$ – деяка опорна множина. Набір ознак NP_{p_a} назвемо тестом, якщо для будь-яких навчальних об'єктів sp'_a, sp''_a, які належать до різних класів, виконана рівність $BN(sp'_a, sp''_a, NP_{pa}) = 0$. Інакше кажучи, тест – це набір ознак, за якими відрізняються будь-які два об'єкти з різних класів.

Тут доречно зауважити, що в наш час найбільш агресивним способом перевірки ефективності СЗІ ІКС від НСД є тест на проникнення. Під час таких заходів у хід ідуть усі можливі способи подолання механізмів захисту ІКС, які тільки можуть використовувати порушники політики безпеки. Результати тестів на проникнення аналізуються, що дозволяє підвищити ефективність СЗІ, а також усунути знайдені уразливості. У країнах Євросоюзу й США проведення тестів на проникнення – одна з найважливіших процедур підвищення ІБ підприємства або корпорації в цілому. У ряді держав модель тесту на проникнення регламентується органами, відповідальними за ліцензування й атестацію у сфері захисту інформації [1-15].

Нехай ΩMI_T – деяка сукупність тестів. Якщо сукупність опорних множин алгоритму складається з тестів, то очевидно, що такий алгоритм є коректним при будь-яких позитивних значеннях параметрів $po_{sp_{ai}}$ $i = 1, 2, ..., PA,$ і $po_{NP_{pa}}$, $NP_{p_a} \in \Omega MI$.

Якщо набір ознак $NP_{p_{a1}}$ – тест, то будь-який набір ознак $NP_{p_{a2}}$ такий, що $NP_{p_{a1}} \subset NP_{p_{a2}}$, також є тестом. При цьому якщо об'єкти близькі за $NP_{p_{a2}}$, то вони будуть близькі й за $NP_{p_{a1}}$. Якщо ж два об'єкти близькі за набором стовпців $NP_{p_{a1}}$, то вони не завжди будуть близькі за $NP_{p_{a2}}$. У цьому сенсі більш короткі тести мають більшу інформативність, і розумно обмежувати довжину тестів (тобто набори ознак) або будувати тупикові тести.

Набір ознак NP_{p_a} назвемо тупиковим тестом, якщо виконані такі дві умови:

1) NP_{p_a} є тестом (тобто набором ознак, що дозволяють виявити загрози ІБ систем);

2) будь-яка власна підмножина набору NP_{p_a} не є тестом.

Іншими словами, тупиковим тестом є набір невкорочуваних ознак, за яким будь-які два ОВН з різних класів загроз інформації $B_{p_a k1}, B_{p_a k2}$ відрізняються один від одного.

Нехай кожна ознака p_{axj}, $j = 1,2,...,n$ має кінцеву множину припустимих значень PA.

Нехай $NP_{p_a} = \{ p_{ax_{j1}},...,p_{ax_{jr}} \}$ – це деякий набір ознак, а $sp_a = (\alpha p_{a1}, \alpha p_{a2},..., \alpha p_{an})$ – об'єкт із навчальної вибірки. Фрагмент $(\alpha p_{aj1},..., \alpha p_{ajr})$ опису об'єкта sp_a позначимо через (sp_a, NP_{p_a}).

Кожний тест NP_{p_a} породжує множину фрагментів описів об'єктів вигляду (sp_{ai}, NP_{p_a}), $i = 1,2,..., PA$, де sp_{ai} – ОВН, причому кожний із цих фрагментів зустрічається в якомусь класі й не зустрічається в інших. Таким чином, якщо при побудові алгоритмів ДПРЗ ІБ перейти від розгляду опорних множин ознак до аналізу фрагментів описів об'єктів, то можна будувати менш обережні й при цьому коректні процедури.

Нехай NP_{p_a} – деякий набір з r_{p_a} різних ознак вигляду $NP_{p_a} = \{ p_{ax_{j1}},..., p_{ax_{jr}} \}$, $\sigma_{DOP} = (\sigma_{DOP_1},..., \sigma_{DOP_r})$, а σ_{DOP_i} – припустиме значення ознаки $p_{ax_i}, i = 1,2,...,r_{p_a}$. Набір σ_{DOP_i} і є ЕК, породженим ознаками з NP_{p_a}. Близькість об'єкта $sp_{an} = (\alpha p_{a1}, \alpha p_{a2},..., \alpha p_{aMI})$ з PA й ЕК $\sigma_{DOP} = (\sigma_{DOP_1},..., \sigma_{DOP_r})$, породженого набором ознак NP_{p_a}, будемо оцінювати величиною:

$$BN(\sigma_{\mathrm{DOP}}, sp_a, NP_{pa}) = \begin{cases} 1, & \text{якщо } \alpha \mathrm{p}_{\mathrm{j_{ti}}} = \sigma_{DOP\,ti} \quad \textit{при} \quad \mathrm{ti} = 1,2,..., \mathrm{r_{p_a}}, \\ 0 & \textit{у протилежному випадку.} \end{cases} \qquad (2.10)$$

Множину усіх ЕК, породжених наборами ознак з $\{p_{ax1},...,p_{axn}\}$, позначимо через MC. Таким чином, $MC=\{(\sigma_{DOP}, NP_{pa})\}$,

де $NP_{pa} \subseteq \{p_{ax1},...,p_{axn}\}$, $NP_{p_a} = \{p_{ax_{j1}},..., p_{ax_{jr}}\}$,

$\sigma_{DOP} = (\sigma_{DOP_1},...,\sigma_{DOP_r})$, $\sigma_{DOP_i} \in NP_{p_{aj}}$, $при\ i = 1,2,...,r_{p_a}$.

Кожний алгоритм AL, що розпізнає загрозу інформації алгоритм для кожного класу KL, $KL \in \{KL_1,...,KL_l\}$, будує деяку підмножину $MC^{AL}(KL)$ множини MC.

Позначимо $MC^{AL} = \bigcup_{j=1}^{l} MC^{AL}(KL_j)$.

Розпізнавання об'єкта sp_{an} здійснюється на основі обчислення величини $BN(\sigma_{DOP}, sp_a, NP_{pa})$ для кожного елемента (σ_{DOP}, NP_{pa}) множини $MC^{AL}(KL), KL \in \{KL_1,...,KL_l\}$, тобто за кожним елементом множини $MC^{AL}(KL)$ здійснюється процедура обчислення оцінки $\Gamma(sp_a, KL)$ приналежності об'єкта sp_a до класу KL. Таким чином, кожний розпізнавальний алгоритм AL з розглянутої групи визначається множиною ЕК $MC^{AL}(KL)$ і способом обчислення $\Gamma(sp_a, KL)$ оцінки.

У загальному випадку ЕК $\sigma_{DOP} = (\sigma_{DOP_1},...,\sigma_{DOP_r})$, породжений ознаками з NP_{pa}, може мати одну з трьох наступних властивостей:

1) кожний фрагмент виду (sp'_a, NP_{pa}), де $sp'_a \in KL$, збігається з $\sigma_{DOP} = (\sigma_{DOP_1},...,\sigma_{DOP_r})$;

2) не всі, а лише деякі фрагменти виду (sp'_a, NP_{pa}), де $sp'_a \in KL$, збігаються з $\sigma_{DOP} = (\sigma_{DOP_1},...,\sigma_{DOP_r})$;

3) жоден фрагмент виду (sp'_a, NP_{pa}), де $sp'_a \in KL$, не збігається з $\sigma_{DOP} = (\sigma_{DOP_1},...,\sigma_{DOP_r})$.

Перша ситуація в СЗІ зустрічається вкрай рідко, тому працювати з наборами значень ознак, для яких виконується властивість 1, не представляється можливим. Істотна відмінність в інформативності наступних двох властивостей полягає в тому, що властивість 2 характеризує лише деяка підмножина ОВН з KL, а властивість 3 – усі об'єкти з KL. Отже, у разі, коли важливо розглядати клас KL ізольовано від інших класів, напрошується висновок про більшу інформативність таких наборів значень ознак, для яких виконана властивість 3. У зазначеному випадку аргументом за віднесення розпізнаваного об'єкта sp_a до класу KL більш природно вважати ситуацію, коли набір значень ознак не присутній у всіх об'єктів із класу KL та не присутній в об'єкта sp_a.

У класичних моделях методика побудови ЕК σ_{DOP_i} для класу KL заснована на побудові σ_{DOP_i} – покриттів матриць, утвореної описами ОВН кожного класу KL. Використання подібних моделей [10] дозволяє трохи знизити обчислювальні витрати у випадку, коли$|KL| < |\overline{KL}|$, наприклад, при великій кількості KL у ІКС.

Відповідно до вихідного переліку окремих завдань, вирішуваних у межах захисту ІКС, розглядається перше з таких завдань - спроба НСД до інформації. Виходячи з певногї, заздалегідь визначеної множини ознак $\{p_{ax_{j1}},...,p_{ax_{jr}}\}$, проводиться визначення (вимір) даних ознак. Із використанням алгоритму автоматизованої класифікації загроз і уразливостей за відміченим набором ознак визначається клас поточних загроз ІБ – KL ($\{p_{ax_{j1}},...,p_{ax_{jr}}\} \to KL_i, i = 1,2,...r_{p_a}$). Із використанням того або іншого механізму прийняття рішень обирається найкраща або, принаймні, раціональна альтернатива, тобто реалізується відповідність – $KL_i \to UR_j, i = 1,2,...AU_j$, де AU_j- число можливих альтернатив (керуючих впливів), на етапі вирішення розглянутого завдання протидії певному класу загроз для ІБ ІКС.

Запропоновано метод виділення типових об'єктів sp_a на основі проведення процедури ковзного контролю, який полягає в такому.

З навчальної вибірки виключається один об'єкт sp_{a_i}, $i \in \{1,2,...,PA\}$. За підвибіркою, що залишилася, $\{sp_{a_1},...,sp_{a_{PA}}\} \setminus sp_{a_i}$ будується розпізнавальний алгоритм. Далі цей алгоритм застосовується для розпізнавання об'єкта sp_{a_i}. Об'єкт sp_{a_i} вважається типовим для свого класу, якщо побудована ДПРЗ працює правильно, і нетиповим, якщо алгоритм відніс його до іншого класу або відмовився від розпізнавання. Описана процедура повторюється для всіх ОВН.

Нехай вибірка ОВН розбита на базову й контрольну підвибірки. За базовою побудуємо множину представницьких наборів. Надамо кожному побудованому представницькому набору якусь вагу, яка обчислюється за контрольною підвибіркою.

Нехай $p\omega$ – представницький набір класу $KL, KL \in \{KL_1,...,KL_l\}$, породжений парою (sp'_a, NP_{pa}), де sp'_a – об'єкт із базової вибірки, і нехай $\delta n(KL, \ p\omega)$ – число об'єктів sp_{a_i} (тобто цілей порушника) у контрольній вибірці, за яких представницький набір «голосує правильно», $\delta n(\overline{KL}, p\omega)$ – число об'єктів у контрольній вибірці, за яких він «голосує неправильно». Тоді як функцію значності ЕК – $vop_{(sp'_a, NP_{pa})}$ можна розглядати функції:

$$\mathrm{vop}_1(sp'_a, NP_{pa}) = \delta \mathrm{n}(\mathrm{KL}, \ \mathrm{p}\omega),$$

$$\mathrm{vop}_2(sp'_a, NP_{pa}) = \frac{1 + \delta \mathrm{n}(\mathrm{KL}, \ \mathrm{p}\omega)}{1 - \delta \mathrm{n}(\overline{\mathrm{KL}}, \mathrm{p}\omega)}.$$

Приналежність об'єкта sp_{a_i} класу KL будемо оцінювати величиною:

$$\Gamma(sp_a, KL) = \frac{1}{|MC^{AL}(KL)|} \cdot \sum_{(sp'_a, NP_{pa}) \in MC^{AL}(KL)} vop_{(sp'_a, NP_{pa})} \cdot (1 - BN(sp_a, sp'_a, NP_{p_a})). \quad (2.11)$$

Як інформативну значущість ознаки p_{axj} будемо розглядати величину:

$$\mathrm{IZ}_{\mathrm{p\,axj}} = \frac{\sum_{\substack{(sp'_a, NP_{pa}) \in \mathrm{MC}^{\mathrm{AL}}(KL) \\ p_{axj} \in NP_{pa}}} \mathrm{vop}_{(sp'_a, NP_{pa})}}{\sum_{\substack{(sp'_a, NP_{pa}) \in \mathrm{MC}^{\mathrm{AL}}(KL) \\ p_{axj} \in NP_{pa}}} \mathrm{vop}_{(sp'_a, NP_{pa})}}. \quad (2.12)$$

Моделювання різних варіантів інформативності значень ознак спроб НСД до ІКС виконано в наступному параграфі роботи.

2.3. Конструювання дискретних процедур інтелектуального розпізнавання загроз інформаційно-комунікаційному середовищу із використанням апарату логічних функцій

У даному розділі роботи викладені основні принципи конструювання ДПРЗ ІБ із використанням апарата логічних функцій, що дозволить на практиці створювати ефективні аналітичні, схемотехнічні та програмні рішення СЗІ для ІКС.

Розглянемо ситуацію, коли об'єкти з досліджуваної множини *PA* описані ознаками, кожна з яких ухвалює значення з множини $\{0, 1,..., k_{p_a} - 1\}$.

Введемо такі позначення:

$\mathrm{PA}_{\mathrm{mn}}^{\mathrm{k_{pa}}}$, $k_{p_a} \geq 2$ – множина усіх матриць уразливостей і загроз із елементами $m \times n$ з $\{0, 1,..., k_{p_a} - 1\}$;

$E_k^{r_{p_a}}$, $k_{p_a} \geq 2, r_{p_a} \leq n$ – множина усіх k_{p_a} -*их* наборів довжини r_{pa};

$\mathrm{Q_p}(\sigma_{DOP}), \sigma_{DOP} \in E_k^{r_{p_a}}, \sigma_{DOP} = (\sigma_{\mathrm{DOP_1}},...,\sigma_{\mathrm{DOP_r}}), p \in \{1,2,...,r_{p_a}\}$ – множина усіх таких наборів виду $\beta_1,...,\beta_r$ в $E_k^{r_{p_a}}$ для яких $\beta_p \neq \sigma_{DOP_p}$ і $\beta_j = \sigma_{DOP_j}$ при $j \in \{1,2,...,r_{pa}\} \setminus \{p\}$;

$CU(LU,\sigma_{DOP})$ – множина усіх пар виду (HU,σ_{DOP});

HU - σ_{DOP} - покриття матриці $LU \in PA_{mn}^k$;

$BU(LU,\sigma_{DOP})$ – множина усіх пар виду (HU,σ_{DOP});

$SU(LU,\sigma_{DOP})$ – сукупність усіх σ_{DOP} – підматриць матриці LU.

Вважаємо:

$$CU(LU) = \bigcup_{r_{pa}=1}^{n=MI} \bigcup_{\sigma_{DOP} \in E_k^r} CU(LU,\sigma_{DOP}), \tag{2.13}$$

$$BU(LU) = \bigcup_{r_{pa}=1}^{n=MI} \bigcup_{\sigma_{DOP} \in E_k^r} BU(LU,\sigma_{DOP}), \tag{2.14}$$

$$SU(LU) = \bigcup_{r_{pa}=1}^{n=MI} \bigcup_{\sigma_{DOP} \in E_k^r} CU(LU,\sigma_{DOP}). \tag{2.15}$$

Очевидно, що найбільший інтерес у наших дослідженнях становить асимптотика типових значень чисел $|CU(LU)|$ *u* $|SU(LU)|$, а також оцінка типового значення $|SU(LU)|/|BU(LU)|$. Виявлення типової ситуації буде пов'язане із твердженням – для майже всіх матриць LU із PA_{un}^k при $n = MI \to \infty$ виконана властивість β, тобто частка тих матриць із PA_{mn}^k, для яких з ε–точністю виконана властивість β, буде прагнути до 1 і одночасно $\varepsilon \to 0$ при $n = MI \to \infty$.

У наших дослідженнях уважаємо, що $PA_{mn}^{k} = \{LU\}$ – це простір елементарних подій, у якому кожна подія LU відбувається з ймовірністю $1/|PA_{mn}^{k}|$. Відповідно математичне очікування випадкової величини $\hbar(LU)$ будемо позначати через $M\hbar(LU)$, а дисперсію – через $D\hbar(LU)$.

Оцінимо, скількома способами можна побудувати матрицю з $PA = PA_{(n\nu_1, m\omega_1, \sigma_{DOP\,1})} \cap PA_{(n\nu_2, m\omega_2, \sigma_{DOP_2})}$. Спочатку виберемо ті елементи, які розташовані на перетинанні рядків з номерами з $n\nu_1$ і стовпців з номерами з $m\omega_1$. Це можна зробити $(k_{p_a} - 1)^{r_{p_a}}$ способами. Потім обираємо елементи, розташовані на перетинанні рядків з номерами з $n\nu_2$ і стовпців з номерами з $m\omega_2$, враховуючи, що *ab* з них розташовані одночасно на перетинанні рядків з номерами з $n\nu_1$ і стовпців з номерами з $m\omega_2$, ($(k_{p_a} - 1)^{l-a}$ – способів). Довільно довизначаємо рядки матриці з номерами з $n\nu_1 \cup n\nu_2$ ($k_{p_a}^{(r_{p_a}+l-a)MI + ab - r_{p_a}^2 - l^2}$ – способів). Обираємо інші рядки довільно ($k_{p_a}^{PA\cdot MI - (r_{p_a}+l-a)MI}$ – способів). Зі сказаного випливає необхідність оцінки для $|PA|$.

Покажемо, що майже для всіх матриць загроз і уразливостей $LU \in PA_{mn}^{k}$ при $n = MI \to \infty$ справедливим є $|SU_1(LU)| = 0$.

На підставі лем [10]:

1. Якщо $n^{\alpha} \le m \le k^{n^{\beta}}$, $\alpha > 1, \beta < 1$, то має місце

$$PA\ \eta_1(LU) \approx PA\ \eta_2(LU) \approx$$

$$\approx \sum_{r_{p_a} \in \lambda_1} CU_m^{r_{p_a}} \cdot CU_n^{r_{p_a}} \cdot r_{p_a}(k_{p_a} - 1)^{r_{p_a}} \cdot k_{p_a}^{r_{p_a} - r^2_{p_a}}$$

при $n = MI \to \infty$;

2. Якщо $n^{\alpha} \le m \le k^{n^{\beta}}$, $\alpha > 1, \beta < 1$, то має місце

$$\frac{D\hbar\eta_2(LU)}{PA(\eta_2(LU))^2} \to 0 \quad \text{при} \quad n = MI \to \infty,$$

де $\eta_{(nv,m\omega)}(LU, \sigma_{DOP})$ – випадкова величина рівна 1, якщо $LU \in PA_{(nv,m\omega,\sigma_{DOP})}$ й рівна 0 а якщо ні, то маємо:

$$PA\eta_3(LU) = \sum_{r_{p_a} \ge r_{pa1}} \sum_{\substack{nv \in V_r^m \\ m\omega \in W_r^n}} \sum_{\sigma_{DOP} \in E_k^r} P(\eta_{(nv,m\omega)}(LU, \sigma_{DOP}) = 1), \tag{2.16}$$

де $nv \in V_{r_{pa}}^m$, $mw \in W_{r_{pa}}^n$, $\sigma_{DOP} \in E_k^{r_{pa}}$;

$P(\eta_{(nv,m\omega)}(LU, \sigma_{DOP}) = 1)$ – ймовірність того, що $\eta_{(nv,m\omega)}(LU, \sigma_{DOP}) = 1.$

Отже, якщо: $| PA_{mv,m\omega,\sigma_{DOP}} | = (k_{p_a} - 1)^{r_{p_a}} \cdot k_{p_a}^{mn - r_{p_a}^2}$,

то одержимо:

$$PA\eta_3(LU) = \sum_{r_{p_a} \ge r_{p_a 1}} CU_{n=MI}^{r_{p_a}} \cdot CU_m^{r_{p_a}} \cdot r_{p_a}!(k_{p_a} - 1)^{r_{p_a}} \cdot k_{p_a}^{r_{p_a} - r^2{}_{p_a}}. \tag{2.17}$$

У зв'язку з тим, що при $r_{p_a} \ge r_{p_a 1}$

$$CU_{n=MI}^{r_{p_a}} \cdot CU_m^{r_{p_a}} \cdot r_{p_p}!(k_{p_a} - 1)^{r_{p_a}} \cdot k_{p_a}^{r_{p_a} - r^2{}_{p_a}} \le \frac{(mn)^{r_{p_a}}}{r_{p_p}!} \cdot r^{2 \cdot r_{p_a} - r^2{}_{p_a}} \le$$

$$\le \left(\frac{k^2{}_{p_a} \cdot e}{r_{p_a}}\right)^{r_{p_a}}, \tag{2.18}$$

то при досить великому $n = MI \to \infty$ будемо мати:

$$\sum_{r_{p_a} \ge r_{p_a 1}}^{n=MI} CU_{n=MI}^{r_{p_a}} \cdot CU_m^{r_{p_a}} \cdot r_{p_p}!(k_{p_a} - 1)^{r_{p_a}} \cdot k_{p_a}^{r_{p_a} - r^2{}_{p_a}} \le$$

$$\le n\left(\frac{k^2{}_{p_a} \cdot e}{\log_{k_{p_a}} nm}\right)^{\log_{k_{p_a}} nm} \to 0. \tag{2.19}$$

Отже, $| BU(LU) | \le CU_{n=MI}^{[\log_{k_{pa}} nm]} mn \le \frac{n^{\log_{k_{pa}} mn} \cdot mn}{[\log_{k_{pa}} mn]!}$.

Також маємо:

$$| SU(LU) | \ge \sum_{r_{pa} \in \hat{\lambda}_1} \frac{(nm)^{r_{pa}-1}}{r_{pa}!}(1-\frac{r_{pa}}{n})^{r_{pa}} \cdot (1-\frac{r_{pa}}{m})^{r_{pa}} \ge$$

$$\ge \sum_{r_{pa} \in \hat{\lambda}_1} \frac{(nm)^{r_{pa}-1}}{[\log_k mn]!}, \qquad (2.20)$$

де $r_{p_{a1}} = \frac{1}{2} \cdot \log_{k_{pa}} mn - \frac{1}{2} \cdot \log_{k_{pa}} \log_{k_{pa}} mn - \log_{k_{pa}} \log_{k_{pa}} \log_{k_{pa}} n$;

$\hat{\lambda}_1$ – інтервал.

Звідси:

$$\frac{| SU(LU) |}{| BU(LU) |} \ge \frac{(mn)^{r_{pa1}-2}}{n^{\log_{k_{pa}} mn}} \ge n^{(\alpha+1)(r_{pq1}-2)-\log_{k_{pa}} mn} \to \infty \qquad (2.21)$$

при $n = MI \to \infty$.

Таким чином, коли число рядків матриці, у якій перераховані уразливості (викликані різними класами загроз ІБ) по порядку більше за число стовпців, то майже завжди величина $| SU(LU) |$ по порядку більше за величину $| BU(LU) |$.

Далі розглянемо базові принципи конструювання ДПРЗ ІБ із використанням апарату логічних функцій.

ЕК (σ_{DOP}, NP_{pa}), де $\sigma_{DOP} = (\sigma_{DOP_1}, ..., \sigma_{DOP_r})$, NP_{pa} – набір ознак з номерами $j_1, ... j_{r_{pa}}$ поставимо у відповідність елементарну кон'юнкцію $\Re = p_{axj_1}^{\sigma_{DOP_1}} ... p_{axj_{r_{pa}}}^{\sigma_{DOP_{r_{pa}}}}$.

Якщо $sp_a = (\alpha p_{a1}, ..., \alpha p_{aMI})$ – об'єкт із множини *PA,* то, мабуть, $BN(\sigma_{DOP}, sp_a, NP_{pa}) = 1$ тоді й тільки тоді, коли $(\alpha p_{a1}, ..., \alpha p_{aMI}) \in NI_{\Re}$, де $NI_{\Re}$ – інтервал істинності елементарної кон'юнкції $\Re$.

Покажемо, що побудова множини ЕК класу KL_l для розглянутих вище моделей, зводиться до знаходження припустимих і максимальних кон'юнкцій для характеристичної функції класу KL_l, тобто такої двозначної логічної функції, яка на ОВН з KL_l і $\overline{KL_l}$ приймає різні значення.

Процедура розпізнавання загрози для певної мети, тобто об'єкта $sp_a = (\alpha p_{a1},...,\alpha p_{aMI})$, здійснюється на підставі розрахунків за побудованими елементарними кон'юнкціям. Тут найбільш економічним є використання алгоритму розрахунків кон'юнкцій за покриттями класу. Характеристична функція класу загроз ІБ KL_l – певна логічна функція F_{KL}, що ухвалює значення 0 на описах об'єктів $sp_{an} = (\alpha p_{an1},..., \alpha p_{anMI})$ з KL_l і значення 1 на інших наборах з E_{KL}^{MI}, тут E_{KL}^{MI} – множина усіх наборів довжини r_{p_a}. Покриттю класу KL_l відповідає припустима для $F_{\overline{KL}}$ кон'юнкція, тупиковому покриттю – максимальна для $F_{\overline{KL}}$ кон'юнкція. Припустима (максимальна) кон'юнкція $\Re$ визначає приналежність об'єкта $sp_{an} = (\alpha p_{an1},...,\alpha p_{anMI})$ класу KL_l, якщо $(\alpha p_{a1},..., \alpha p_{aMI}) \notin NI_{\Re}$.

Завдання побудови скороченої ДНФ функції звичайно зводиться до завдання побудови скороченої ДНФ функції F_{KL}, що приймає значення 0 на наборах з $B_{F_{\overline{KL}}}$ і значення 1 на інших наборах E_{KL}^{MI}. Після побудови ДНФ функції $F_{\overline{KL}}$ вилучимо з неї кон'юнкції $\Re$, що не володіють властивістю $NI_{\Re} \cap A_{F_{KL}} \neq 0$.

Побудувати скорочену ДНФ логічної функції можна також шляхом перетворення кон'юнктивної форми вигляду $D_1 \wedge D_2 \wedge ... \wedge D_u$, де $D_i = p_{ax1}^{\overline{\beta_{i1}}} \vee p_{ax2}^{\overline{\beta_{i2}}} \vee ... \vee p_{axMI}^{\overline{\beta_{iMI}}}, i = 1,2,..,mu$ реалізує функцію F_{KL}, β_{iMI} – елементи набору $B_{F_{\overline{KL}}}$.

Скористаємося рівністю $\overline{p_{ax}{}^{\alpha}} = \bigvee\limits_{\beta_i \neq \alpha_i} p_{ax}{}^{\beta}$.

Тоді кон'юнктивна форма набуває вигляду $D^*{}_1 \wedge D^*{}_2 \wedge ... \wedge D^*{}_u$,

де $D^*_i = \bigvee_{\iota \neq \beta_{i1}} p^{\eta}_{ax1} \vee \bigvee_{\iota \neq \beta_{i2}} p^{\eta}_{ax2} \vee ... \vee \bigvee_{\iota \neq \beta_{iMI}} p^{\eta}_{axMI}, i = 1,2,..,u.$

Побудова множини ЕК для модельованого класу загроз ІКС, які підлягають розпізнаванню, зводиться до такого: 1) задається характеристична функція; 2) будується ДНФ, що реалізує цю функцію. Найбільшу складність становить побудова ДНФ із максимальних кон'юнкцій (скороченої ДНФ) характеристичної функції; 3) обчислюється припустима (максимальна) кон'юнкція $\Re$, що визначає приналежність об'єкта до певного класу загроз ІБ.

Питання застосування відповідних характеристичних функцій у повному обсязі в межах даного дослідження не розглядалося, оскільки для кожного класу цілей нападу на інформацію, можна знайти різні математичні підходи до опису характеристичних функцій [10 та ін.]. Використання нечітких змінних у нашій моделі суттєво підвищує гнучкість програми класифікації й дозволяє реалізувати функціональність, необхідну для оперування такими сутностями, як КНІ. Кожному значенню ознаки об'єкта ставиться у відповідність нечітка змінна, здатна відобразити ступінь упевненості експерта (програми) у значенні якої-небудь ознаки.

У таблиці 2.1 наведено приклад навчальної матриці для уразливостей та, відповідно, що призводять до одержання прав доступу володіння до довірених суб'єктів ІКС.

При побудові матриць використовувалося поняття інформативності ознаки. Інформативними ознаками називається корисна для ДПРЗ інформація, отримана з вхідної інформації. Інформативність ознаки означає, наскільки дана ознака характеризує стан ІБ об'єкта ІКС, тобто наскільки від нього залежить постановка задачі розпізнавання загрози.

Таблиця 2.1

Фрагмент навчальної матриці для уразливостей, що призводять до одержання прав доступу володіння до довірених суб'єктів ІКС

Уразливості, що призводять до одержання прав доступу володіння до довірених суб'єктів	Показники (ознаки)		
	Кількість зафіксованих інцидентів (p_{axl})	Інформа-тивність значення ознаки ($-1 \le IZ_{p_{axj}} \le 1$)	Існує тенденція до збільшення кількості загроз ІБ, викликана уразливістю?
пропущені в системних функціях ОС, застосовуваних у вузлах ІКС, механізми перевірки на наявність прав використання адміністра-тивних повноважень	$p_{axl} \le 10$ $IZ_{p_{axj}} = 0{,}15$	$IZ_{p_{axj}} = 0{,}2$	так $IZ_{p_{axj}} = 0{,}4$
помилки в ПЗ, наприклад, що реалізує прикладні й системні процеси ОС	$1 \le p_{axl} \le 10$ $-0{,}1 \le IZ_{p_{axj}} \le 0{,}6$	$IZ_{p_{axj}} = 0{,}5$	немає $IZ_{p_{axj}} = 0{,}25$
відсутність аутентифікації й авторизації користувачів при доступі до ресурсів вузлів ІКС (довіра)	$p_{axl} \ge 10$ $0{,}1 \le IZ_{p_{axj}} \le 1$	$IZ_{p_{axj}} = 1$	немає $IZ_{p_{axj}} = 0{,}4$
помилки в реалізації, конфігуруванні та використанні ІКС	$p_{ax1} \ge 5$ $-0{,}2 \le IZ_{p_{axj}} \le 1$	$IZ_{p_{axj}} = 0{,}8$	немає $IZ_{p_{axj}} = 0{,}2$
порушення політики зберігання паролів для доступу в ІКС	$p_{axl} \le 5$ $0{,}2 \le IZ_{p_{axj}} \le 1$	$IZ_{p_{axj}} = 0{,}4$	так $IZ_{p_{axj}} = 0{,}8$
можливість одержання або зміни деяких параметрів ЛОМ, що дозволяють ідентифікувати використовуване на вузлах системне й прикладне ПЗ, установлені відновлення ОС і зареєстрованих у них користувачів	$p_{axl} \le 5$ $-0{,}2 \le IZ_{p_{axj}} \le$	$IZ_{p_{axj}} = 0{,}7$	Так $IZ_{p_{axj}} = 0{,}5$

Для завдання інтелектуального розпізнавання загроз використовувався інформаційний підхід, згідно з яким інформація ознаки розглядається, як достовірне розходження між класами ОВН в просторі ознак. Оцінкою інформативності служить величина $IZ_{p_{axj}}$ – площа одного розподілу ознак $\{p_{axl},..., p_{axn}\}$, не спільна з площею іншого розподілу цієї ж ознаки [10].

Зрозуміло, чисельні значення ознак можуть змінюватися залежно від ступеня оснащеності ІКС комплексами СЗІ на кожному з рубежів захисту.

Використання нечітких змінних дозволяє не тільки описувати нечіткі характеристики КНІ або спроб НСД, але й проводити нечіткий логічний висновок після побудови класифікаційних правил за отриманим деревом рішень.

З наборів $A_{F_{KL}}$ і $B_{F_{KL}}$ складемо, відповідно, матриці LU_1 й LU_2 (тут $LU \in PA_{paMI}^{k_{pa}}$) вигляду:

$$\begin{bmatrix} \alpha_{11} & \alpha_{12} \dots \alpha_{1MI} \\ \alpha_{21} & \alpha_{22} \dots \alpha_{2MI} \\ \dots & \dots \quad \dots \quad \dots \\ \alpha_{mv1} & \alpha_{mv2} \dots \alpha_{mvMI} \end{bmatrix} \quad \text{та} \quad \begin{bmatrix} \beta_{11} & \beta_{12} \dots \beta_{1MI} \\ \beta_{21} & \beta_{22} \dots \beta_{2MI} \\ \dots & \dots \quad \dots \quad \dots \\ \beta_{mu1} & \beta_{mu2} \dots \beta_{muMI} \end{bmatrix}.$$

У цьому випадку елементарна кон'юнкція на наборі ознак $p_{ax\ j_1}^{\sigma_{DOP_1}} \dots p_{ax\ jr}^{\sigma_{DOP_r}}$ є припустимою для функції F_{KL} тоді й тільки тоді, коли набір стовпців з номерами $j_1,\dots,\ j_{r_{pa}}$, є $\sigma_{DOP_1},\dots,\sigma_{DOP_{r_{pa}}}$ – покриттям матриці LU_2. Елементарна кон'юнкція на наборі ознак $p_{ax\ j_1}^{\sigma_{DOP_1}} \dots p_{ax\ jr}^{\sigma_{DOP_r}}$ є максимальною для функції F_{KL} тоді й тільки тоді, коли набір стовпців з номерами $j_1,\dots,\ j_{r_{pa}}$ є тупиковим покриттям матриці LU_2.

Раніше в [10] вивчалися випадки, коли число рядків у матриці *PA* по порядку менше за число стовпців. Було показано, що в цьому випадку величина $|BN(LU)|$ при $MI \to \infty$ асимптотично збігається з величиною $|sp_a(LU)|$ й по порядку менше за число покриттів (тут $LU \in PA_{paMI}^{k_{pa}}$). Різноманітність розглянутих ознак у межах одного класу призводить до завдання опису різних класів ознак і співвідношень між ними. Зауважимо, що множина *MIQ* має

$p_{ax\ j}^{\sigma_{DOP}}$ – ознаку, якщо $MIQ \cap p_{ax\,j}^{\sigma_{DOP}} \neq 0$. Назвемо ознаку тривіальною, якщо $MIQ \cap p_{ax\,j}^{\sigma_{DOP}}$ – одноелементна множина. Множина MIQ не має ознаки $p_{ax\ j}^{\sigma_{DOP}}$, якщо $MIQ \cap p_{ax\,j}^{\sigma_{DOP}} = 0$.

Коли зазначені вище етапи будуть завершені, можна переходити до робіт із формування моделі загроз ІБ для ІКС на базі отриманих класифікаторів. Вихідними даними для моделювання будуть класи й підкласи уразливостей, загроз і цілей, а також множини засобів для реалізації нападів на ІКС і категорій (класів) зловмисників (див. табл. 2.2).

Метод складання вирішального правило $gov(p_{axi})$ для визначення стану S_R ІКС у випадку загрози для ІБ базується на процедурі аналізу критичності окремих елементів, що входять до складу ІКС, і визначається етапами:

1) для кожного вузла $um^* \subset EUM^*$ визначаються довірені користувачі, що володіють правом доступу володіння до кожної сутності (наприклад, інформаційному масиву – $M_{k\,inf}$);

2) множини EUM^*, SDN й функція IR не змінюються на всіх траєкторіях графа станів системи.

3) для одержання зловмисником (суб'єктом-порушником) SDN_x права володіння щодо суб'єкта SDN_i йому, як правило, потрібно одержати доступ не тільки до сутності EUM_z^*, але й доступ на запис/читання до деякої сутності $eum_l \in EUM^*$, що є інтерфейсом або портом деякого суб'єкта-процесу $pro \in SDN$, що здійснює надання прав доступу SDN_i на основі даних у сутності EUM^*.

4) сутності EUM_z^* і eum_l є асоційованими із суб'єктом pro_r^m; сутності eum_l й pro_r^m, як правило, розміщені на одному вузлі мережі, а сутності EUM_z^* й EUM_y^* можуть бути розміщені на різних вузлах КС.

Таблиця 2.2

Класи та підкласи загроз ІБ для ІКС

Клас загрози	Підклас загрози	Описові ознаки, обумовлені експертами
за зоною враження	для ІКС	[1]
	загрози для предметних зон ІКС	[2]
	загрози для соціальної системи, що йдуть від ІКС	[7]
за ознакою їх зв'язку з інформаційним середо-вищем соціальної системи	зовнішні	[9]
	внутрішні - йдуть від соціальної системи та її елементів	[12]
	внутрішньосистемні – йдуть від ІКС	[10]
за силою впливу на зону враження	1) руйнівні, 2) дестабілізуючі, 3) паралізуючі	[7]
за формою вираження й ступенем соціальної небезпеки	1) колізії, 2) конфлікти, 3) проступки, 4) злочини, 5) аварії, 6) катастрофи	[6]
за метою реалізації загрози	Порушення: конфіденційності; цілісності; доступності	[6]
за принципом впливу	із використанням доступу в АС	[9]
	із використанням прихованих каналів	[6]
	із використанням мереж загального користування	[6,14]
за характером впливу	активні	[10,15]
	пасивні	[9,15]
через появу використовуваної помилки захисту	неадекватність політики безпеки	[10,11]
	помилки керування системою захисту	[9,15]
	помилки проектування системи захисту	[9,15]
	помилки кодування	[9,15]
за способом впливу на об'єкт нападу на інформацію	безпосередній вплив на об'єкт нападу на інформацію	[9,15]
	вплив на систему дозволів	[9,15]
	опосередкований вплив	[9,15]
за способ ом впливу	в інтерактивному або пакетному режимі,	[9,15]
	ІС у цілому, об'єкти ІС, суб'єкти ІС	[9,15]

	канали передачі даних	[9,15]
за використовуваними засобами нападу на інформацію	із використанням: штатного ПЗ; розробленого ПЗ	[9,15]
	із використанням: загальнодоступних ТЗ; спеціальних (спеціально спроектованих і створених) технічних засобів	[9,15]
за станом об'єкта нападу на інформацію	при зберіганні об'єкта	[9,15]
	при передачі об'єкта	[9,15]
	при обробці об'єкта	[9,15]

5) вирішальне (розв'язувальне) правило $gov(p_{axi})$, яке описує стани ІКС, можна представити в такому вигляді (див. табл. 2.3).

Фактори, що впливають на вибір рішення щодо формування вирішального правила $gov(p_{axi})$, представлені у вигляді лінгвістичних змінних у базі знань для розпізнавання загроз ІБ (табл. 2.4), для яких вибрані універсальні множини та терми. Для формалізації лінгвістичних змінних була вибрана дзвіноподібна модель функції належності, яка має найменше число параметрів, що зменшує розмірність задачі підбору цих параметрів при навчанні [10].

Для кожного із співвідношень дерева висновку побудовані нечіткі бази знань (див. табл. 2.5), які представляють сукупність нечітких правил «ЯКЩО-ТОДІ», що визначають взаємозв'язок між вхідними та вихідною змінними, див. табл. 2.4, 2.5, 2.7. За нечіткими базами знань складені логічні рівняння. Правило активується, якщо істинність його умови більша за нуль. В базах знань (див. табл. 2.5) процедура агрегування умов в правилах виконується за допомогою нечітких логічних операцій – нечіткої кон'юнкції, нечіткої диз'юнкції, нечіткої відмови, та ін.

Таблиця 2.3

Вирішальне правило $gov(p_{axi})$ для визначення стану ІКС у випадку загрози для ІБ

Правило	Вихідний стан ІКС S_R	Результуючий стан ІКС S'_R
$gov(p_{axi}) =$ $= (SDN_x,$ $SDN_y,$ $EUM^*_z,$ $eum_l,$ $pro^m_r)$	$SDN_x, SDN_y,$ $pro^m_r \in EUM^*,$ $eum_l,$ $EUM^*_z \in EUM,$ $eum_l \in pro^m_r,$ $(SDN_x, eum_l,$ $write_r / read_r \in RDN),$ $EUM^*_z \in SDN_y$ і *або* $SDN_x = SDN_y,$ *або* $(EUM^*_z,$ $SDN_x,$ $write_m / read_m) \in MIF, KL,$ $MC \in AL(KL)$	$S_R = S'_R,$ $EUM^* = EUM'^*,$ $ADN = ADN',$ $IR = IR',$ $MIF = MIF',$ $RDN' = RDN'(SDN_x,$ $SDN_y),$ $KL \in (KL_1,...KL_l),$ $MC \in AL$ $(KL \in (KL_1,...KL_l))$

Таблиця 2.4

База знань для розпізнавання загроз ІБ

Класи загроз ІБ		Атрибути		Ознаки $\{p_{ax1},\ldots,p_{axn}\}$	Інформативність значення ознаки $IZ_{p_{axj}}$	Універсум	Терми для лінгвістичної оцінки
Можливі загрози ІБ ІКС	**Відомі загрози**	KL_1	Відмова в обслуговуванні елементів ІКС або АСК, SCADA, HMI, PLC та ін.	1-не працюють штатні компоненти; 2-не працюють штатні компоненти; 3-ін.	$0 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	некритичний (нкр), критичний (кр)
		KL_2	Викрадення інформації або компонентів ІКС, ІС або АСК, SCADA, HMI, PLC	1- об'єктивні ознаки (наприклад, поява конфіденційної інформації у ЗМІ); 2-субуєктивні ознаки; 3-ін.	$-0,5 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлені (в), частково невиявлені (чв), невиявлені (нв)
		KL_3	Привласнення особистості у ІКС, ІС або АСК, SCADA, HMI, PLC	1- об'єктивні ознаки (наприклад, зафіксовані спроби роботи під чужим логіном); 2-субуєктивні ознаки; 3-ін.	$0,1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлені (в), частково невиявлені (чв), невиявлені (нв)
		KL_4	Модифікація інформації у ІКС, ІС або АСК, SCADA, HMI, PLC	1-зміна контенту; 2-зміна структури документів; 3-ін.	$-0,5 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлена (в), частково невиявлена (чв), невиявлена (нв)
		KL_5	Злам пароля користувача	1 – виявлення кейлоггера; 2-атипова поведінка користувача; 3-ін.	$0,1 \le IZ_{p_{axj}} \le 1$	$[0, N_a]$	зафіксовані СЗІ (зф), незафіксовані СЗІ(нф)

Продовження таблиці 2.4

Можливі загрози ІБ ІКС	**Відомі загрози**		Вірусна атака на ІКС, ІС або АСК, SCADA, HMI, PLC	1-незвичайні прояви в роботі ІС; 2-зміни заданої в передостанньому сеансі роботи з ЕОМ структури файлової	$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлена (в), частково невиявлена (чв), невиявлена (нв)
		KL_7	Пошук залишкової інформації	1- об'єктивні ознаки; 2-субуєктивні ознаки; 3-ін.	$-0{,}5 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлені (в), частково невиявлені (чв), невиявлені (нв)
		KL_8	Несанкціонований запуск ПЗ ІС або АСК, SCADA, HMI, PLC	1-незвичайні прояви в роботі ПЗ ІС; 2- атипова поведінка ПЗ; 3-ін.	$0{,}1 \le IZ_{p_{axj}} \le 1$	$[0, N_a]$	зафіксовані СЗІ (зф), незафіксовані СЗІ(нф)
		KL_9	Зміна конфігурації у СЗІ ІС або АСК, SCADA, HMI, PLC	1-чужорідне тіло; 2-сторонні сигнали; 3-ін.	$0{,}1 \le IZ_{p_{axj}} \le 1$	$[0, N_a]$	зафіксовані СЗІ (зф), незафіксовані СЗІ(нф)
		KL_{10}	Несанкціоноване знищення даних ІС або АСК, SCADA, HMI, PLC	1- об'єктивні ознаки; 2-субуєктивні ознаки; 3-ін.	$-1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	некритичні (нкр), критичні (кр)
		KL_{11}	Несанкціоноване відкриття файлів ІС або АСК, SCADA, HMI, PLC	1- об'єктивні ознаки; 2-субуєктивні ознаки; 3-ін.	$-1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлені (в), частково невиявлені (чв), невиявлені (нв)
		KL_{12}	Зміна конфігурацій ІС або АСК, SCADA, HMI, PLC	1-нові модулі; 2-сторонне ПЗ; 3-ін.	$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлені (в), частково невиявлені (чв), невиявлені (нв)
		KL_{13}	Зміна конфігурації обладнання систем управління інфра-структурою	1-чужорідне тіло; 2-сторонні сигнали; 3-ін.	$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлена (в), частково невиявлена (чв), невиявлена (нв)

Продовження таблиці 2.4

Можливі загрози ІБ ІКС	**Відомі загрози**		KL_{14}	Зміна конфігурації обладнання систем управління АПК (SCADA)	1-чужорідне тіло; 2-сторонні сигнали; 3-сліди установки; 4-ін.	$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлені (в), частково невиявлені (чв), невиявлені (нв)
			KL_{15}	Зміна конфігурації обладнання систем навігації	1-Рівень сигналу; 2-Однаковий рівень сигналу від різних супутників; 3-Візуальні ознаки відхилення від маршруту; 4-Ін.	$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлена (в), частково невиявлена (чв), невиявлена (нв)
			KL_{16}	Зміна конфігурації обладнання систем оповіщення та відеоспостереження	1-чужорідне тіло; 2-сторонні сигнали; 3-сліди установки; 4-ін.	$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлені (в), частково невиявлені (чв), невиявлені (нв)
			KL_{15}	Зміна конфігурації обладнання систем зв'язку GSM-R, VSAT, GSM та ін.	1-Рівень сигналу; 2-Однаковий рівень сигналу від різних супутників; 3-Ін.	$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлена (в), частково невиявлена (чв), невиявлена (нв)
			KL_{15}	Зміна конфігурації обладнання систем управління зв'язку	1-Рівень сигналу; 2-Однаковий рівень сигналу від різних супутників; 3-Ін.	$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлена (в), частково невиявлена (чв), невиявлена (нв)
			KL_{16}	Порушення доступності ІМ та ПЗ ІКС	1-не працюють штатні компоненти ІС АПК (ІМ та ПЗ);	$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	некритичний (нкр), критичний (кр)
			KL_{17}	Резервний атрибут			[0,1], у. о.	некритичний (нкр), критичний (кр)
			KLO_{21}	Поширення зловмисного коду під client-side рівень	1-не працюя відповідне ПЗ; 2-ін.	$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлена (в), частково невиявлена (чв), невиявлена (нв)

Продовження таблиці 2.4

	Невідомі загрози	**Описані**	KIO_{22}	Можливий злам систем GPM-R та здійснення атак типу «DoS/DDoS» на системи SCADA АПК		$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлена (в), частково невиявлена (чв), невиявлена (нв)
			KIO_{23}	Злам VSAT систем та здійснення атак типу «DoS/DDoS»		$0{,}1 \le IZ_{p_{axj}} \le 1$	[0,1], у. о.	виявлена (в), частково невиявлена (чв), невиявлена (нв)
		Неописані	KLN_{31} KLN_{3m}					
Стани системи								
S_{IK_1} - встановлене ПЗ та оновлення до нього; S_{IK_2} - в системі присутні мережеві сервіси; S_{IK_3} - система підтримує багатозадачність; S_{IK_4} - підтримка багатокористувацького режиму; S_{IK_5} - встановлені пристрої введення / виводу; S_{IK_6} - наявність пристроїв «гарячої заміни»; S_{IK_7} - наявність зовнішніх каналів зв'язку; S_{IK_8} - наявність системи відеоспостереження з'єднаної із корпоративною системою; S_{IK9} - наявність системи супутникової навігації з'єднаної із корпоративною системою; S_{IK10} - наявність ЗЗІ.								
Методи протидії								
$D_{сзі_1}$ - ідентифікація і аутентифікація; $D_{сзі_2}$ - блокування безконтрольного доступу; $D_{сзі_3}$ - захист від вірусів; $D_{сзі_4}$ - контроль цілісності даних; $D_{сзі_5}$ - знищення залишкових даних; $D_{сзі_6}$ - захист ПЗ та ІМ від дослідження; $D_{сзі_7}$ - резервування інформації; $D_{сзі_8}$ - відновлення і самовідновлення компонентів ІКС; $D_{сзі_9}$ - перевірка сертифіката безпеки; $D_{сзі_{10}}$ - блокування запуску ПЗ; $D_{сзі_{11}}$ - криптографічний захист; $D_{сзі_{12}}$ - ін.								

Таблиця 2.5

База знань у вигляді правил

Правила			
PR1	*IF (*$\mathrm{KL}_1 \vee S_{IK_2} \vee S_{IK_7}$*) THEN* $\mathrm{D}_{\text{ззі}_2}$	*PR10*	*IF (*$\mathrm{KL}_9 \vee S_{IK_1}$*) THEN* $\mathrm{D}_{\text{ззі}_8}$
PR2	*IF (*$\mathrm{KL}_1 \vee S_{IK_2} \vee S_{IK_7}$*) THEN* $\mathrm{D}_{\text{ззі}_2}$	*PR11*	*IF* $(\mathrm{KL}_{10} \vee S_{IK_5})$ *THEN* $\mathrm{D}_{\text{ззі}_7}$
PR3	*IF (*$\mathrm{KL}_2 \vee S_{IK_4} \vee S_{IK_7}$*) THEN* $\mathrm{D}_{\text{ззі}_2}$	*PR12*	*IF (*$\mathrm{KL}_{11} \vee S_{IK_4} \vee S_{IK_7}$*) THEN* $\mathrm{D}_{\text{ззі}_2}$
PR4	*IF (*$\mathrm{KL}_3 \vee S_{IK_4}$*) THEN* $\mathrm{D}_{\text{ззі}_8}$	*PR13*	*IF (*$\mathrm{KL}_{11} \vee S_{IK_5}$*)THEN* $\mathrm{D}_{\text{ззі}_7}$
PR5	*IF (*$\mathrm{KL}_4 \vee S_{IK_1}$*) THEN* $\mathrm{D}_{\text{ззі}_4}$	*PR14*	*IF (*$\mathrm{KL}_{11} \vee S_{IK_4} \vee S_{IK_7}$*) THEN* $\mathrm{D}_{\text{ззі}_2}$
PR6	*IF (*$\mathrm{KL}_5 \vee S_{IK_3}$*) THEN* $\mathrm{D}_{\text{ззі}_6}$	*PR15*	*IF (*$\mathrm{KL}_5 \vee S_{IK_7}$*) THEN* $\mathrm{D}_{\text{ззі}_2}$
PR7	*IF (*$\mathrm{KL}_6 \vee S_{IK_1}$*) THEN* $\mathrm{D}_{\text{ззі}_3}$	*PR16*	IF $(\mathrm{KL}_5 \vee \mathrm{KL}_8 \vee \mathrm{KL}_{11} \vee S_{IK_5} \vee \vee S_{IK_7})$ THEN $\mathrm{D}_{\text{ззі}_{10}}$
PR8	IF $(\mathrm{KL}_7 \vee S_{IK_4} \vee S_{IK_5} \vee \vee S_{IK_6} \vee S_{IK_7})$ THEN $\mathrm{D}_{\text{ззі}_5}$	*PR17*	IF $(\mathrm{KL}_3 \vee \mathrm{KL}_9 \vee \mathrm{KL}_4)$ *THEN* $\mathrm{D}_{\text{ззі}_9}$
PR9	*IF (*$\mathrm{KL}_8 \vee S_{IK_4}$*) THEN* $\mathrm{D}_{\text{ззі}_1}$	*PR18* ... PR_{nm}	*Інші*

Перевагами методі інтелектуального розпізнавання загроз (ДПРЗ) ІБ є, див. табл. 2.6:

1) одержання функції класифікації з мінімальним рівнем помилки класифікації;

2) можливість використання лінійних класифікаторів для роботи з нелінійно поділюваними даними;

3) можливість роботи з різнорідними складноструктурованими даними за рахунок використання різних функцій;

4) у разі зміни структури аналізованих даних достатньо замінити тільки використовуване правило $gov(p_{axi})$, без заміни самого алгоритму розпізнавання загрози ІБ.

Таблиця 2.6

Порівняльний аналіз методів розпізнавання загроз ІБ

Алгоритми які використовуються	дерево рішень	нейроні мережі	Байєсівські мережі довіри	методі нечіткої логіки
точність	+/-	-	-	-
можливість пояснення	-	+	+	-
швидкодія	+/-	+	+/-	+/-
здатність до навчання	+/-	+/-	-	-
«+» – наявність труднощів; «-» – відсутність труднощів; «+/-» – часткові труднощі				

Запропонована математична модель інтелектуального розпізнавання із використанням апарату логічних функцій була реалізована в середовищі MATLAB 7 [5] і надалі включена в підсистеми моделювання ІБ ІКС (див. розд. 4).

Для кожного виду загроз КНІ, а також, каналів відпливу інформації, яка циркулює на підприємствах АПК, складалася навчальна вибірка з 10–95 об'єктів (sp_{an}), розбитих на класи, як показано в табл. 2.7. Для кожного класу кількість ознак варіювалася від 3 до 9 (див. табл. 2.5, 2.7). Інформативність ознаки змінювалася в діапазоні від –1 до +1. Для оцінки ефективності процедур розпізнавання використовувався метод ковзного контролю.

На рис. 2.4 показаний приклад структури блоку «Chart» (Matlab та Simulink), що дозволяє оцінити взаємодія розглянутих загроз ІБ.

Рис. 2.4. Структура блоку «Chart»

Таблиця 2.7

Найбільш поширені типи КНІ у ІКС

№	Загрози нападу на інформацію (*TAI* - дані про сценарії атак (нападів на інформацію) у вигляді	Пояснення	Кількість розглянутих ознак
1	Несанкціоноване використання точок доступу дистанційного технічного обслуговування (ТО)	Точки доступу для ТО - спеціально створені зовнішні входи в мережу ІКС, які часто бувають недостатньо безпечними.	9
2	Мережеві атаки через КІС або мережу	Офісні ІТ (зазвичай) підключені до мережі декількома способами. У більшості випадків існують мережеві канали зв'язку між офісами і мережею АСК, ІС.	9
3	Атаки на стандартні компоненти, що використовуються в ІКС	Стандартні компоненти ІТ (готові комерційні продукти), такі як системне програмне забезпечення, сервер додатків або БД містять недоліки і уразливості.	9
4	DoS/DDoS –атаки на ІКС	Атака типу «відмова в обслуговуванні» може найважливіших ресурсів, викликати збій систем, наприклад, для того, щоб перервати роботу ІКС та ін.	9
5	Людська помилка або саботаж персоналу ОЦ	Навмисні дії - як з боку внутрішніх, так і зовнішніх порушників - являють собою масову загрозу для всіх об'єктів ІКС.	6
6	Запуск вірусу через знімний носій або зовнішні пристрої	Використання знімних носіїв і мобільних ІТ-компонентів пов'язане з високим ризиком зараження вірусом (наприклад, випадок з Stuxnet).	6
7	Читання і запис новин в мережі ICS	Більшість компонентів контролю в даний час використовує протоколи незашифрованого тексту, таким чином, дана комунікація залишається незахищеною. Це спрощує читання і введення команд управління.	7
8	Атаки на ERP через протокол HARD	Нападники використовують команди до датчиків, що дозволяє обмінюватися інформацією між інтелектуальними приладами контролю руху.	6
9	Атаки на компоненти мережі	Нападники можуть маніпулювати компонентами мережі.	9
10	Атаки систем SCADA технічного обслуговування, обліку, заправки та ін.	Створення активних радіозавад у зоні роботи SCADA системи. Для створення постійних перешкод використовуються генератори «білого шуму», що працюють в тій же смузі частот, що і SCADA-системи.	5

Продовження табл. 2.7

11	Атаки на HMI	Несанкціонований доступ до Інтернет-інтерфейсу диспетчера з мобільного пристрою може здійснюватися у разі використання відкритих бездротових мереж чи мереж зі слабкою системою автентифікації.	3
12	Атака з направлена на ініціювання «відмови в обслуговуванні» або доступу до АСК, HDI, PLC.	Перехоплення кадрів сенсорних вузлів з метою підміни MAC адрес джерел і приймачів, що призводить до відмов у роботі SCADA-системи. Підміна центрального координатора з метою зміни адресного простору в конфігурації сенсорної мережі.	6
13	Атака спрямована на зміну прошивок, драйверів і ПЗ контролерів (PLC) і сенсорних вузлів (RFD - Зниження функції пристроїв).	Атака ведеться шляхом сканування PLC і сенсорних вузлів для визначення можливостей зміни встановленої ОС, прошивки, драйверів і ПЗ контролерів.	5
14	Впровадження та / або підміна вузлів («атака воронки»)	Проводиться шляхом підміни вузлів, відповідальних за збір та ретрансляцію даних у мережі (FFD - Повністю багатофункціональний пристрій) з метою перехоплення і пере направлення мережевого трафіку, що створює загрозу для роботи SCADA-системи та HDI.	9
15	Атаки на бортове навігаційно-зв'язне обладнання	Фізичний вплив на блок терміналу або його вміст проводиться з метою пошкодити або тимчасово відключити вузли терміналу. Програмне вплив полягає у безпосередньому підключенні зловмисника до приладу за допомогою дата-кабелю або ж у віддаленому підключенні до каналу зв'язку на проміжку сервер-пристрій. Такий вплив вимагає наявність спеціального обладнання і навичок у зловмисника і дозволяє змінювати внутрішні налаштування приладів і навіть їх прошивку. Таке вторгнення практично неможливо виявити, але можна попередити, використовуючи фільтрацію по IP і MAC адресами і надійні методи автентифікації.	6
16	Компрометація вузла збору даних ІКС, SCADA, HDI, PLC	Впроваджений в рамках атаки на мережу вузол постійно генерує і розсилає квитанції підтвердження з адресою реального вузла збору, що збільшує ймовірність отримання даної квитанції вузлами-джерелами раніше квитанції від реального координатора мережі і, як наслідок, подальшої передачі кадру даних на впроваджений вузол.	9

Продовження табл. 2.7

17	Підміна маршрутизатора (FFD вузла) в SCADA, HDI для порушення коректної роботи алгоритмів маршрутизації.	Створення «помилкового» тунелю. Установка фільтрів. Зміни маршрутів.	7
18	Знімання інформації з клавіатури, принтера, сканера, за рахунок побічного випромінювання терміналі АРМ у складі ІКС та ін.	Використовується з метою отримання інформації по встановлене ПЗ та дії операторів, крадіжку інформації, тощо.	9
19	Відплив інформації з дисплея по електромагнітному каналу	Використовується з метою отримання інформації по встановлене ПЗ та дії операторів.	3
20	Візуальний з'їм інформації з компонентів ІС засобами вбудованих камер мобільних телефонів і т.п.	Використовується з метою отримання інформації по встановлене ПЗ та дії операторів та ін.	3
21	Комп'ютерні віруси, експлойти і т. п.	Можуть використовувати уразливості в ПЗ та проведення атаки на обчислювальну систему. Метою атаки може бути як захоплення контролю над системою (підвищення привілеїв), так і порушення її функціонування (DoS-атака).	9
22	Програмно-апаратні закладки у АСК, ІКС	Вносять довільні спотворення в коди програм. Переміщують фрагменти інформації з одних областей пам'яті в інші. Навмисно змінюють інформацію.	7
23	Отримання доступу до пристроїв ІКС які використовують протокол	Метою атаки може бути як захоплення контролю над системою (підвищення привілеїв), так і порушення її функціонування (DoS-атака).	8
24	Отриманням доступу незареєстрованого користувача до ІС, АСК з боку віддаленої машини або права локального супер-користувач	Метою атаки може бути як захоплення контролю над системою (підвищення привілеїв), так і порушення її функціонування (R2L, U2R, DoS-атаки).	5
25	Інше		9

На рис. 2.5 - 2.10 показані основні результати, отримані в ході моделювання інформативності значень ознак спроб НСД до інформаційних ресурсів ІКС.

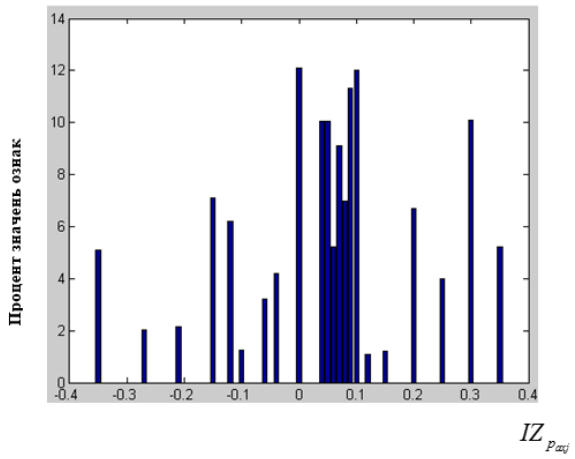


Рис. 2.5. Розподіл типовості значень $IZ_{P_{axj}}$ для можливих каналів відпливу інформації

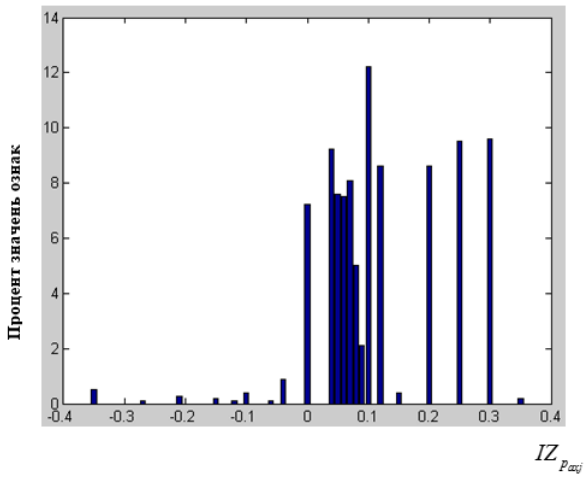


Рис. 2.6. Розподіл типовості значень інформативності ознак для КНІ на інформацію

Дослідження показали, що в моделі «голосування» за представницькими наборами при вирішенні завдань аналізу загроз ІБ ІКС досить обмежитися побудовою представницьких наборів довжини 3. Якщо при побудові алгоритму розпізнавання загроз ІБ додавалися представницькі набори більшої довжини, ефективність алгоритму виявлялася такою ж. При додаванні представницьких наборів меншої довжини ефективність алгоритму знижувалася.

Діаграми показують розподіл ваги значень ознак. Як показано на рис. 2.5, об'єкти з різних класів загроз ІБ важко віддільні один від одного, що і є причиною низької ефективності класичного алгоритму розпізнавання.

На рис. 2.6 бачимо, що в завданні аналізу КНІ у ІКС частина значень ознак мають вагу близьку до нуля, але при цьому багато і таких значень, які мають досить велику вагу, тобто є дуже типовими для одного із класів.

При вирішенні завдань інтелектуального розпізнавання загроз ІБ із використанням представницьких наборів довелося відмовитися від вимоги безвихідності представницького набору, тому що перевірка безвихідності значно знижує швидкість роботи алгоритму. Використовувалися представницькі набори обмеженої довжини. Максимальна довжина набору бралася рівною 3. При меншій максимальній довжині більша частина об'єктів не містила жодного представницького набору. А збільшення максимальної довжини до 4 різко збільшувало час роботи алгоритму. Був отриманий такий результат (див. рис. 2.7 – 2.10).

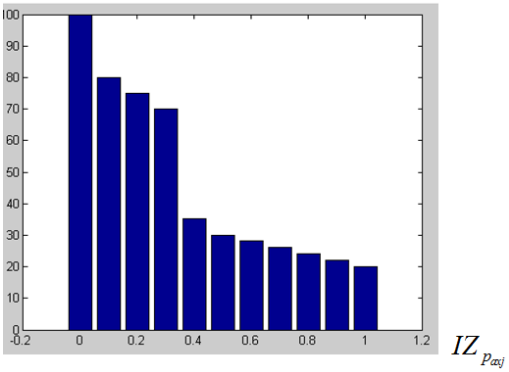


Рис. 2.7. Розподіл інформативності ознак для DoS/DDoS – атаки

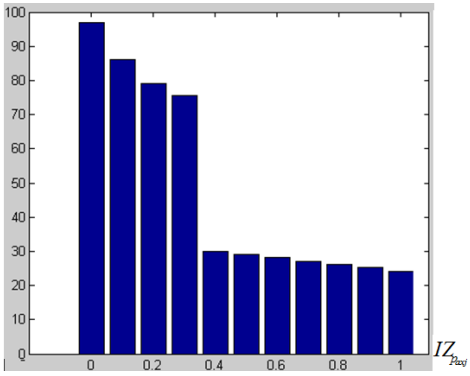


Рис. 2.8. Розподіл інформативності ознак НСД до системи GPS

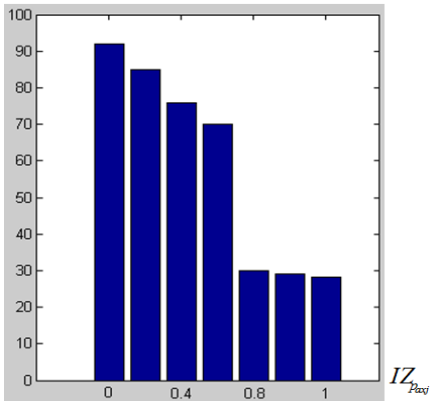


Рис. 2.9. Розподіл інформативності ознак НСД до системи зв’язку з/т GSM-R

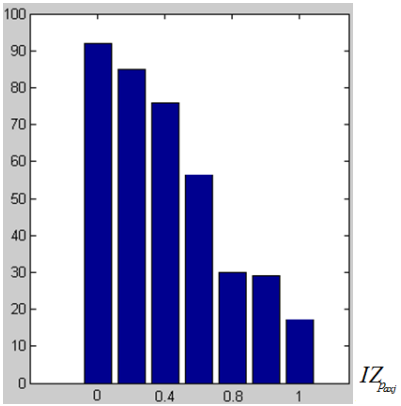


Рис. 2.10. Розподіл інформативності ознак НСД до PLC АСК

Якщо розташувати ознаки класу в порядку убування інформативності, то, як правило, у кожному класі є виділена група ознак з великою інформативністю, далі йде деякий розрив, а ознаки, що потім залишилися, вибудовуються в ряд із плавно зменшуваною інформативністю.

Наприклад, у завданні оцінки впливу DoS/DDoS атаки на ІКС [13], модулі e-business, e-logistics, e-cargo [1] та ін., як інформаційні ознаки можна використовувати такі (див. рис. 2.7): зниження пропускної здатності каналу; зміна частотної характеристики; ін.

У завдання оцінки впливу атаки на системи супутникової навігації, найбільш інформативними є наступні ознаки, див. рис. 2.8.

1) Рівень сигналу. Сигнал супутників GPS на поверхні Землі досить слабкий, його рівень – близько –163 дБ*Вт Сигнал, що випромінюється імітатором значно сильніше, що може свідчити про атаку.

2) Однаковий рівень сигналу від різних супутників. GPS-сигнали різних супутників зазвичай сильно відрізняються за рівнем [10].

3) Шум. Підроблений сигнал GPS має дуже низький рівень шуму.

4) Номери супутників.

Для завдання оцінки внутрішніх каналів відпливу інформації через персонал можна використовувати такі ознаки: спроби доступу (без відповідного дозволу) у приміщення, у якому перебуває АРМ або сервери ІКС; випадки роботи співробітника на чужому АРМ ІКС; ін.

Також, виявилося, що інформативність набору значень ознак може суттєво (іноді на порядок) перевищувати вагу ознак, які його становлять. Інакше кажучи, фрагмент, породжений двома ознаками, може більш сильно характеризувати один із класів, ніж значення кожної із зазначених ознак окремо.

На основі запропонованої моделі ДПРЗ, розроблена експертна система (ЕС) для виявлення НСД та складних нападів на інформацію у ІКС, яка працює під операційними системами Windows, див. рис. 2.11.

Для тестування продуктивності ДПРЗ та розробленої ЕС обрана задача зіставлення загрозам відомих методів захисту. База знань із 18 правил розрізняє 22 класу загроз на основі відомих ознак, вхідних і додаткових атрибутів, що описують поточний стан системи (див. табл. 2.4, 2.5, 2.7). База знань використовувалася для ініціалізації ЕС; та генерації тестових даних на основі наявних правил.

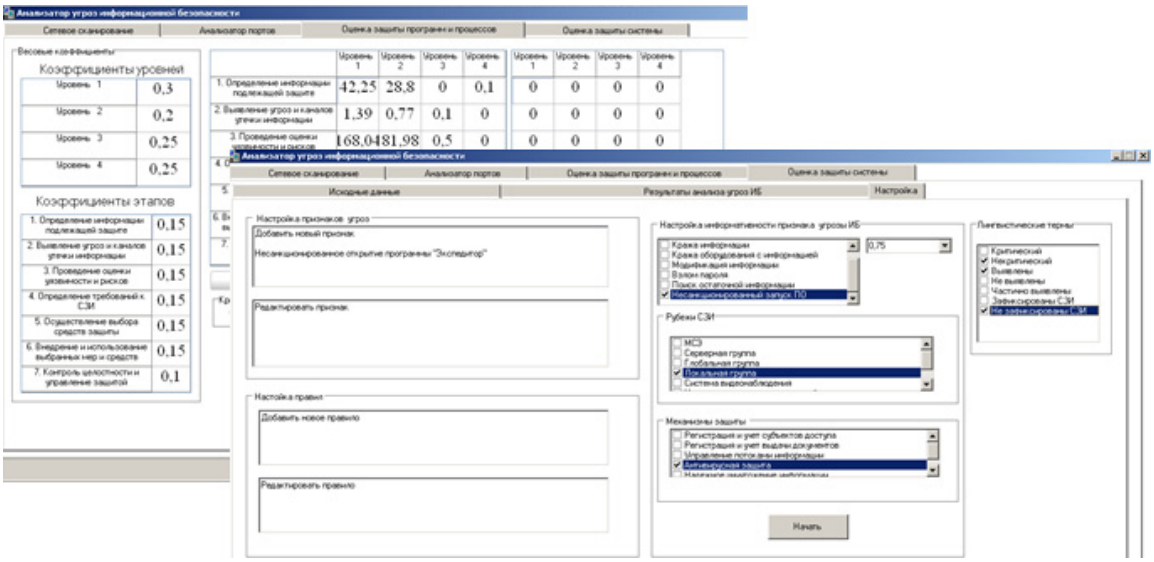


Рис. 2.11. Експертна система для розпізнавання загроз ІБ ІКС

Зроблена оцінка ймовірності $P_{рз}$ надійного розпізнавання загроз для ІБ [107, 225, 225]

$$P_{рз} = \Phi\left(\frac{0{,}5 \cdot \sum_{i=1}^{N_{ра}} \left[1 + \Phi(\mathrm{IZ}_{\mathrm{p_{axj}}} / 2) \cdot \log_2 n_i\right]}{2 \cdot N}\right),$$

де Φ – інтеграл ймовірності [15];

$N_{ра}$ – кількість ознак нападу на інформацію;

n_i – число градацій ознаки нападу на інформацію.

Технічний результат полягає в тому, що стає можливою оптимальна організація множин ознак нападу на інформацію і їх градацій та побудова вирішальних правил $gov(p_{axi})$, в результаті чого підвищується ймовірність розпізнавання, в тому числі і атипових складних КНІ в умовах статистичної недостатності наявних баз ознак у базі знань (див. таб. 2.4, 2.5, 2.7). Результати тестових завдань показані на рис. 2.12 – 2.17.

Розглянувши запропонований метод інтелектуального розпізнавання загроз ІБ ІКС, у наступному розділі роботи зупинимося на особливостях використовування моделі ДПРЗ для розпізнавання складних атак на ІКС. Зокрема, розглянемо модель інтелектуального розпізнавання загроз ІКС в умовах неоднорідних потоків запитів [9].

В таблиці 2.8. наведено порівняння середньої точності розпізнавання загроз для найбільш поширених методів [6].

Таблиця 2.8

Порівняння точності розпізнавання загроз для різних методів

Метод або алгоритм розпізнавання загроз	Середня точність, %
ЕМ алгоритм	77
Дерево рішень	67
Нейроні мережі	80
Метод K-NN	79
Гаусівський класифікатор	89
ДПРЗ+ нечіткі бази знань	85

Якщо знати характерні ознаки несанкціонованих дій (механізми реалізації нападів на інформацію), а саме: присутність повтору певних подій у системі; неправильні або невідповідні встановленим процесам поточні ситуації та команди; використання уразливостей; невідповідні параметри мережного трафіка; непередбачені атрибути; додаткові знання про порушення, – то можна виявити або знизити ризики від реалізації досить складних нападів на інформацію у ІКС.

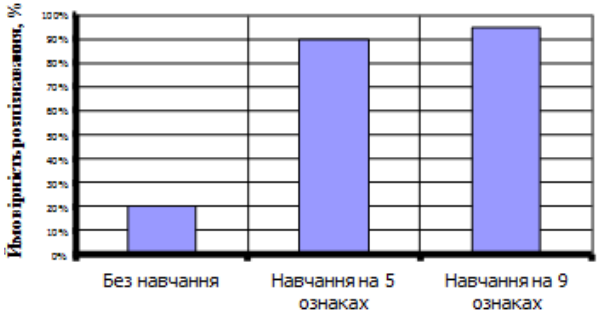


Рис. 2.12. Ймовірність розпізнавання загрози «НСД до відеосервера»

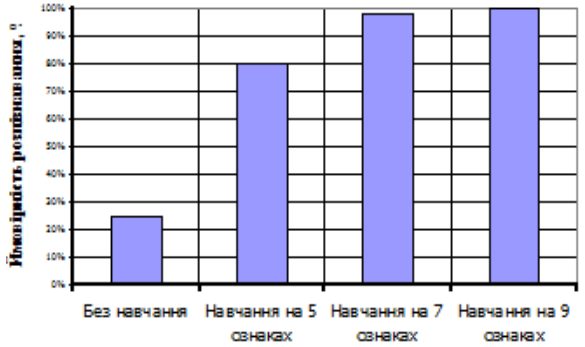


Рис. 2.13. Ймовірність розпізнавання загрози «НСД до пароля користувача»

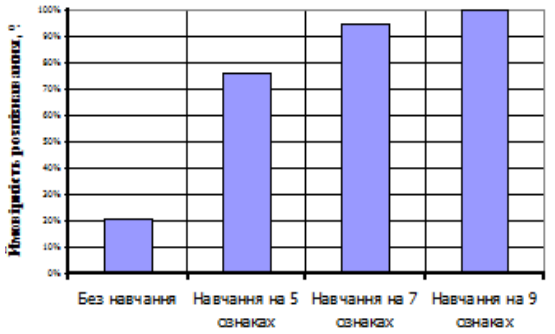


Рис. 2.14. Ймовірність розпізнавання загрози «НСД до ПЗ та БД ІС АРК»

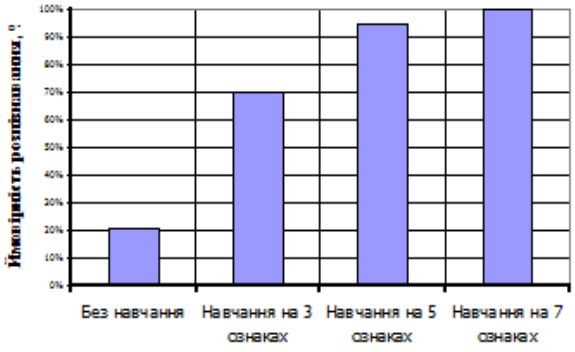


Рис. 2.15. Ймовірність розпізнавання загрози «НСД до систем супутникової навігації»

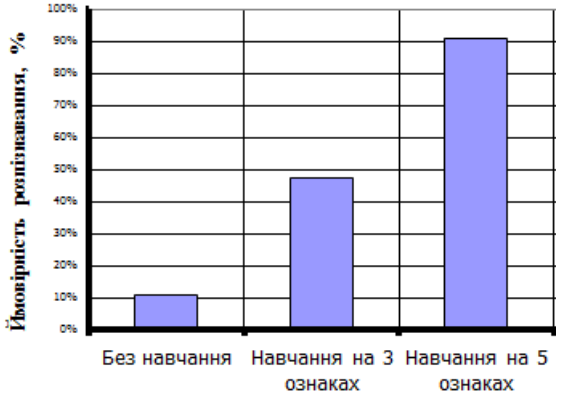


Рис. 2.16. Ймовірність розпізнавання загрози «НСД до PLC АСК рухом ТЗ»

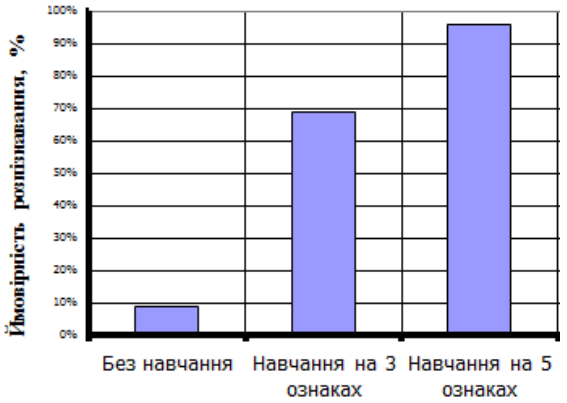


Рис. 2.17. Ймовірність розпізнавання загрози «НСД до засобів сигналізації керування рухом ТЗ»

На транспорті, наприклад, в першу чергу цінується надійність рішень, що забезпечують захист корпоративної мережі. Інформація про переміщення ТЗ дуже важлива і пов'язана з життям людей. Якщо АСК, ІС вийдуть з ладу, диспетчер повинен відразу ж дізнатися про це, щоб встигнути відреагувати.

Варто зазначити, що абсолютна більшість аналітичних робіт в галузі математичного моделювання режимів функціонування АСК, ІС та АС заснована на моделюванні тривалості інтервалів між послідовними входженнями запитів у систему розподілених за показовим законом. Це дозволяє представляти вхідні запити потоками Пуассона [12]. Однак у певних ситуаціях, наприклад, коли постає завдання виходу ІС із нормального режиму роботи, не можна говорити про незалежність надходження запитів на опрацювання, а отже, про адекватну роботу ДПРЗ ІБ. У цьому випадку потоки запитів не є потоками Пуассона [6]. Для потоків такої структури адекватною математичною моделлю є потік Бартлетта [3]. Виходячи із даної обставини, у наступному розділі роботи розглядається робота ІКС у разі втрати заявок внаслідок блокування потоку ДПРЗ ІБ при нападі на інформацію у ІКС, коли втрати виникають через переповнення заявками накопичувача.

2.4. Оцінка показника поточного ризику реалізації загроз інформаційно-комунікаційному середовищу

Інформаційна безпека ІКС вимагає обліку всіх подій, в процесі яких інформація створюється, модифікується, до неї виконується доступ або вона передається. Реалізація даних заходів є вимогою міжнародного стандарту ISO/IEC 27001:2005 «Системи управління інформаційною безпекою. Вимоги».

Необхідність використання систем моніторингу ІБ в ІКС або АСК визначається тим, що застосування звичайних засобів і механізмів захисту інформації виявляється недостатнім, тому що вони виконують лише базові функції і не дозволяють контролювати безпеку функціонування систем в умовах постійних модифікацій нападів на інформацію, оновлення апаратного та

ПЗ. Існуючі системи моніторингу безпеки ІКС не здатні проводити комплексну оцінку дій зловмисників, будувати послідовність реалізованих уразливостей, визначати кінцеву ціль дій зловмисників і оцінювати ризики реалізації загроз для безпеки ІС та АСК, що може привести до повторних успішних атак, а також до фінансових втрат.

Існуючи засоби моніторингу ІБ, наприклад, Intruder Alert, RealSecure Network Sensor [2] та ін., мають певні недоліки:

високий рівень помилок першого і другого роду, тобто помилок не визначення нападу на інформацію, коли воно має місце, та формування ситуації «атака» при її відсутності;

відсутність можливості адаптивно управляти системою ІБ та проводити випереджувальні дії.

Зниження ризиків НСД до інформаційних ресурсів ІКС може бути досягнуте вирішенням завдання оцінки й аналізу поточної небезпеки процесу КНІ в реальному часі, щоб завчасно попередити напад і тим самим запобігти розвитку несприятливого сценарію розвитку даної ситуації. Очевидно, що для врахування впливу великої кількості параметрів нападу на інформацію на ступінь ІБ ІКС, а також їх взаємозв'язки систем в реальному масштабі часу, необхідні спеціальні методи й відповідні організаційні, технічні та програмні засоби.

Аналіз потенційних поточних ризиків дозволяє визначити найбільш актуальні для ІКС загрози та заходу протидії їм, а також оптимізувати вартісні витрати на побудову системи захисту.

Одним із найпоширеніших методів оцінки ризику є метод, заснований на моделі системи «з повним перекриттям», що представляє собою тріаду «загрози – засобу захисту інформації – об'єкти захисту» у вигляді тридольного графа [11].

При проведенні оцінки ризику можна виділити три постановки завдання, що відображають цілі такої оцінки:

1) оцінка ризику на об'єктах ІКС, необладнаних ЗЗІ, з метою з'ясування необхідності створення комплексів СЗІ.

2) оцінка ризику реалізації загроз КНІ з метою модернізації існуючих комплексів СЗІ.

3) оцінка ризику з метою створення нового комплексу СЗІ.

Отже, потрібно розраховувати ризик для всіх трьох постановок завдання проектування. Урахування реальних зв'язків загроз і ресурсів призводить до того, що ризик необхідно визначати з урахуванням значень елементів матриці зв'язків, значення якої рівні 0 – якщо загроза не може впливати на ресурс, і рівні 1 – якщо загроза потенційно може впливати на інформаційний ресурс.

Одне із завдань системи інтелектуального розпізнавання загроз полягає у формулюванні показника поточних ризиків (ППР) небезпеки нападу на інформацію в ІКС та його оцінки в реальному масштабі часу.

Для вирішення даного завдання в роботі передбачається виконати наступне:

1) одержати кількісний показник поточних ризиків (ППР) реалізації загроз ІКС;

2) розробити алгоритми оцінки ППР для компонентів ІКС, що працюють у реальному масштабі часу, з урахуванням поточних значень ДПРЗ ІБ і нових класів загроз.

Поточні ризики та загрози нападу на інформацію або несанкціонованого проникнення є показниками технологічних процесів в ІКС, які можуть набувати різних значень залежно від різних факторів (див. табл. 2.4, 2.5, 2.7). Інтуїтивно зрозуміло, що поточні ризики можуть бути незначними, якщо всі потенційно небезпечні параметри ІКС (наприклад, ІКС або АСК) підтримуються у встановлених межах, або збільшуватися, набуваючи загрозливого характеру, при відхиленні таких параметрів від норми. Тому виникає необхідність описувати ступінь поточної небезпеки ризиків реалізації загроз нападу на інформацію за допомогою деякого кількісного показника, значення якого залежало б від відхилень параметрів, пов'язаних із ІБ.

Уведення такого показника дозволить здійснювати непрямий вимір ступеня поточних ризиків реалізації загрози нападу на інформацію у ІКС та АСК.

Необхідно відмітити, що певна кількісна характеристика небезпеки НСД уже була введена в нормативній документації й у дослідженнях ряду авторів [3 і ін.]. Відповідно до цих джерел, як кількісна характеристика небезпеки проникнення в ІКС розглядається ризик, вимірюваний, як правило, у грошових одиницях, що для критично важливих систем не завжди має першорядне значення.

Ми пропонуємо ввести спеціальний показник для кількісної характеристики ступеня поточної небезпеки КНІ або НСД у ІКС, який може бути розрахований (виміряний) у будь-який момент часу, зокрема, із використанням ДПРЗ ІБ. Результати вимірів показника можуть бути представлені системному адміністраторові (адміністраторові СЗІ) або використані для вирішення інших завдань.

Якщо процеси у ІКС або АСК характеризуються ризиками реалізації загроз ІБ, усі значення яких лежать у зоні припустимих значень ZS_0 (див. рис. 2.18), то поточна ІБ може вважатися нульовою. У разі якщо один або кілька параметрів переходять у зону небезпечних значень ZS_1, то поточна небезпека збільшується, і вона буде зростати з наближенням параметрів до зони критичних значень ZS_2.

Зрозуміло, що поточна небезпека проникнення в ІКС повинна залежати від загального числа загроз інформації – *MI*, що одночасно перебувають у зоні ZS_1, від ступеня наближення кожного параметра до зони ZS_2 і від ступеня впливу кожної загрози на можливість виникнення позаштатної ситуації, наприклад, одержання доступу до ресурсів ІКС.

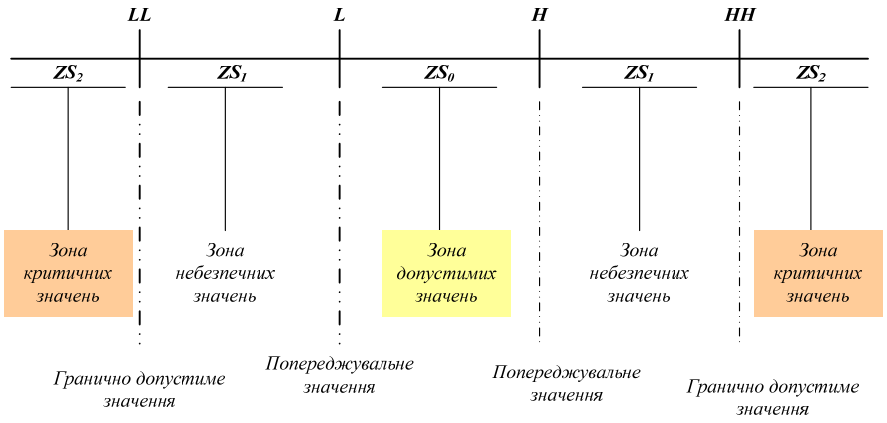


Рис. 2.18. Ризики реалізації загроз ІБ ІКС

Позначимо ППР через $C_{ППР} = C_{ППР}(\bar{X})$, де $\bar{X}_{ППР} = (x_{ППР1},...,x_{ППР\,i},...,x_{ППР\,MI})$ – вектор значень ППР, MI – число загроз інформації.

Показник поточних ризиків $C_{ППР}$ повинен відповідати таким вимогам.

1. Бути скалярною безрозмірною величиною, що змінюється від 0 до 1 ($C_{ППР} = (0 \div 1)$) з урахуванням алгоритму роботи ДПРЗ ІБ.

2. Бути функцією параметрів $x_{ППР\,i}$ $C_{ППР} = f((x_{ППП1},...,x_{ППРi},...,x_{ППРМ})$.

3. Значення $C_{ППР}$ повинні залежати від значень усіх ризиків реалізації загроз для компонентів і процесів в ІКС, коли вони перебувають у зоні небезпечних значень $0 < C_{ППР} < 1$,

$$\textit{якщо} \quad \exists(\mathrm{x}_{\mathrm{ППРi}}) : ZS_i = ZS_1,\ i = 1..MI;$$
$$\textit{або} \quad \exists(\mathrm{x}_{\mathrm{ППРi}}) : (x_{ППРil} > x_{ППРi} > x_{ППРil}) \vee (x_{ППРih} > x_{ППРi} > x_{ППРihh}),$$

де $x_{ППРil}, x_{\mathrm{ППР}\,ih}$ – попереджувальні значення параметрів,

$x_{ППТ\ ill}, x_{ППР\ ihh}$ – гранично припустимі значення параметрів.

4. Значення ППР ІБ ІКС дорівнює нулю, якщо всі параметри інформаційних процесів в ІКС, перебувають у зоні допустимих значень $C_{ППР} = 0$,

$$\textit{якщо} \quad \forall(\mathrm{x}_{\mathrm{ППР\,i}}) : ZS_i = ZS_0,\ i = 1..MI.$$

5. Значення ППР дорівнює одиниці, якщо хоча б один технологічний параметр ІС та АСК (див. табл. 2.4, 2.5, 2.7) перебуває в зоні критичних значень $C_{ППР} = 1$,

$$якщо \qquad \exists(\text{x}_{\text{ППР i}}) : ZS_i = ZS_2, i = 1..MI .$$

6. Значення $C_{ППР}$ повинне бути зростаючою функцією своїх аргументів.

Якщо $C_{ППР1}$ – значення показника $C_{ППР}$ при $x_{ППРi} = x_{ППР}1$,

$C_{ППР2}$ – значення показника $C_{ППР}$ при $x_{ППРj} = x_{ППРj}1$,

і якщо ступінь впливу $x_{ППРi}$ буде менше ніж ступінь впливу $x_{ППРj}$ то $C_{ППР1} < C_{ППР2}$.

7. Значення $C_{ППР}$ повинне зростати зі збільшенням числа загроз та NIS – даних про інциденти з ІБ у зонах ZS_1 і ZS_2.

Якщо $C_{ППР1}$ – значення показника $C_{ППР}$ при $x_{ППРi} = x_{ППРi}1, \quad ZS_i = ZS_1$,

$C_{ППР2}$ – значення показника $C_{ППР}$ при

$$x_{ППРi} = x_{ППРi}1,\ x_{ППРj} = x_{ППРj}1,\ ZS_i = ZS_1, \quad ZS_j = ZS_1\ то\ C_{ППР1} < C_{ППР2}.$$

8. Показник $C_{ППР}$ повинен ураховувати ступінь впливу кожної загрози в межах класу KL_i на можливість виникнення аварійної ситуації, що виникає при атаці на компоненти ІКС.

Якщо $C_{ППР1}$ – значення показника $C_{ППР}$ при $x_{ППРi} = x_{ППР}1$,

$C_{ППР2}$ – значення показника $C_{ППР}$ при $x_{ППРj} = x_{ППРj}1$,

і якщо ступінь впливу $x_{ППРi}$ буде менше, ніж ступінь впливу $x_{ППРj}$,

то $C_{ППР1} < C_{ППР2}$.

9. Значення $C_{ППР}$ повинне бути застосовне в будь-якому режимі функціонування ІКС.

Розрахунки ППР НСД у ІКС проводяться за такою залежністю [1]:

$$C_{ППР}(\overline{X}) = \sqrt{\frac{x^2_{ППР\ 1}}{\prod_{i=2}^{MI}(1 + x^2_{ППРi})} + \sum_{i=2}^{MI} \frac{x^2_{ППРi}}{\prod_{k=i}^{MI}(1 + x^2_{ППРk})}}. \qquad (2.22)$$

Формула обчислення значення $C_{ППР}$ при використанні в ІКС вимагає спеціального алгоритму нормування та впорядкування параметрів MI.

Дослідження алгоритму й основних властивостей ППР проведене за допомогою програмного пакету Mathcad.

Припустимо, що кожний $C_{ППР}$ може мати одну або дві зони небезпечних значень ($H - high$ і $L - low$).

Якщо параметр $x_{ППР\,i}$ $(1 \le i \le MI)$ має одну або дві зони небезпечних значень, то перетворення його поточного значення в нормовану величину θ_i виконується за формулою (2.23):

$$\theta_i = \begin{cases} 1, & якщо \quad x_{ППРi} \le x^{ll}_{ППРi}, \\ \dfrac{x_{ППР\,i} - x^{ll}_{ППР_i}}{x^{l}_{i} - x^{ll}_{i}}, & якщо \quad x^{ll}_{ППРi} < x_{ППРi} < x^{l}_{ППРi}, \\ 0, & якщо \quad x^{l}_{ППРi} \le x_{ППРi} \le x^{h}_{ППРi}, \\ \dfrac{x_{ППРi} - x^{h}_{ППРi}}{x^{hh}_{ППРi} - x^{h}_{ППРi}}, & якщо \quad x^{h}_{ППРi} < x_{ППРi} < x^{l}_{ППРi}, \\ 1, & якщо \quad x_{ППРi} > x^{hh}_{ППРi}, \end{cases} \qquad (2.23)$$

де $x_{ППРi}$ – поточне значення параметра;

$x^{l}_{ППРi}, x^{h}_{ППРi}$ – попереджувальні значення параметра;

$x^{ll}_{ППРi}, x^{hh}_{ППРi}$ – гранично припустимі значення параметра.

Ступінь впливу кожного з факторів на можливість реалізації загрози ІБ при відхиленні цього параметра від норми, задається шляхом ранжирування, тобто присвоєння параметру певного коефіцієнта (рангу). Ранг параметра K_r представляє собою додатну цілочисельну величину: $K_r = 1, 2, \ldots$.

На рис. 2.19 - 2.20 наведені, як приклад, залежності ППР $C_{ППР}$ від значень параметрів $x_{ППРi}$ і θ_i для $MI = 1, 2$ і $K_r = 1, 2$ (наприклад, викрадення ключів для системи зв'язку GSM-R і організація DoS/DDoS атаки на систему управління рухом транспортних засобів).

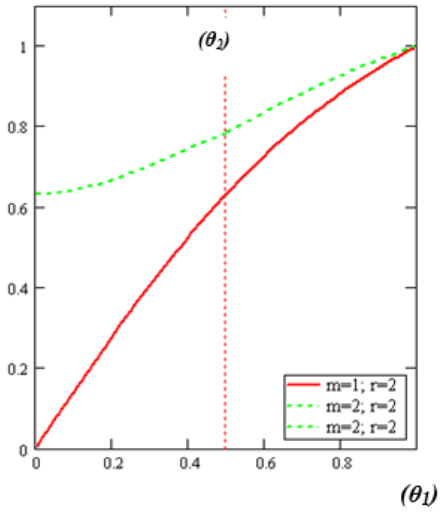


Рис. 2.19. Залежності $C_{ППР}(\theta_1)$ для $MI=1$ і 2 і $Kr_1 = Kr_2 = 2$

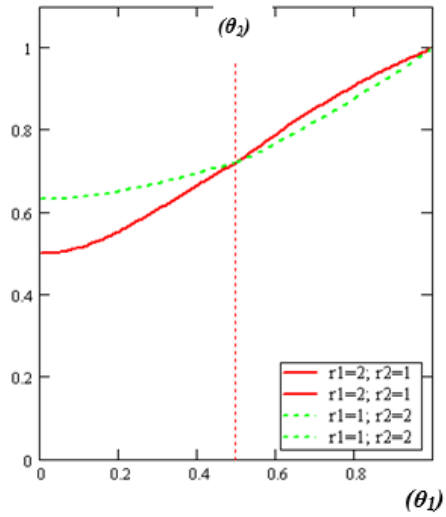


Рис. 2.20. Залежності $C_{ППР}(\theta_1)$ для $MI=2$, $Kr_1 > Kr_2$ і $Kr_1 < Kr_2$

У результаті аналізу цих та інших залежностей, досліджених у роботах [9], зробимо такі висновки:

1. $C_{ППР} = 0$, якщо все θ_i рівні 0.

2. $C_{ППР} = 1$, якщо хоча б один параметр із θ_i рівний 1.

3. $0 < C_{ППР} < 1$, якщо хоча б один параметр із θ_i більше нуля й менше 1 (відповідний параметр x_i перебуває в небезпечній зоні).

4. Якщо при знаходженні одного або декількох параметрів у небезпечній зоні ще один параметр теж потрапляє в небезпечну зону, то показник $C_{ППР}$ зростає.

5. Якщо кількість поточних загроз ІБ $MI=1$ і $K_{r1}=1$, те $C_{ППР} = \theta_1$, тобто показник небезпеки $C_{ППР}$ пропорційний параметру θ_1.

6. Якщо $MI=1$ і $K_{r1}=2$, то залежність $C_{ППР}$ від θ_1 нелінійна. У цьому випадку $C_{ППР}$ більше, ніж при $K_{r1}=1$ для тих самих значень θ_1.

7. Якщо $MI>1$, то всі залежності $C_{ППР}$ від θ_1 нелінійні. При цьому значення показника $C_{ППР}$ тим вище, чим більше параметрів перебувають у небезпечній зоні.

8. Чим вище ранг параметрів, що перебувають у небезпечній зоні, тим вище показник $C_{ППР}$ за інших рівних умов.

Специфіка критично важливих комп'ютерних систем (КВКС) до яких відносяться і ІКС та АСК, така, що вони знаходяться під загро-зою не тільки проникнення звичайних вірусів, але можуть стати об'єктом спрямованих (цільових) атак з боку зловмисників або кібер-терористів.

Серед ризиків ІБ, специфічних для КВКС, можна назвати наступні. По-перше, ризики реалізації загроз КНІ підвищує використання в ІКС та АСК застарілого ПЗ, обладнання та комунікаційних протоколів (для яких спочатку не припускали навіть самої можливості атаки). По-друге, це адміністративні та технологічні труднощі для оновлення ПЗ. По-третє, підключення ізольованій мережі АСК до загальної мережі ІКС або навіть до Інтернету (наприклад, сучасні контролери які зазвичай використовуються АСК рухом на транспорті, можуть бути сполучені безпосередньо або через модем. При підключенні через модем їх часто об'єднують з GPRS/GSM-модемами, що за замовчуванням наділяє пристрій IP – адресою мобільного оператора. При такій конфігурації вони дуже уразливі для атак ззовні. Спеціалізованими утилітами і методами

зловмисник може виявити подібні пристрої та отримати доступ, безпосередньо через канал Ethernet / Industrial Ethernet). По-четверте, це доступ сторонніх компаній до технологічної мережі [3].

Якщо говорити про компоненти АСК, то найбільш уразливими є PLC-контролери, а також SCADA-системи. Типовими загрозами безпеці PLC є жорстко запрограмовані логін і пароль адміністратора, які зазвичай закладаються виробником з метою зручності подальшого обслуговування та підтримки, а також схильність PLC-контролерів до мережевих атак на зразок DoS/DDoS [2].

В системах SCADA, типовими загрозами ІБ є поширене ПЗ для ОС Windows, наприклад, Buffer overflow, DoS/DDoS, SQL Injection.

Оцінка загроз ІБ ІКС та АСК включає дві складові: ситуаційний аналіз і виявлення загроз [13, 14].

Ситуаційний аналіз являє собою детальний аналіз параметрів функціонування апаратно-програмного забезпечення КВКС. При проведенні даного аналізу доцільно згрупувати однотипні дані і оцінювати їх окремо по кожній групі. Приклад такого аналізу показаний на рис. 2.21–2.22.

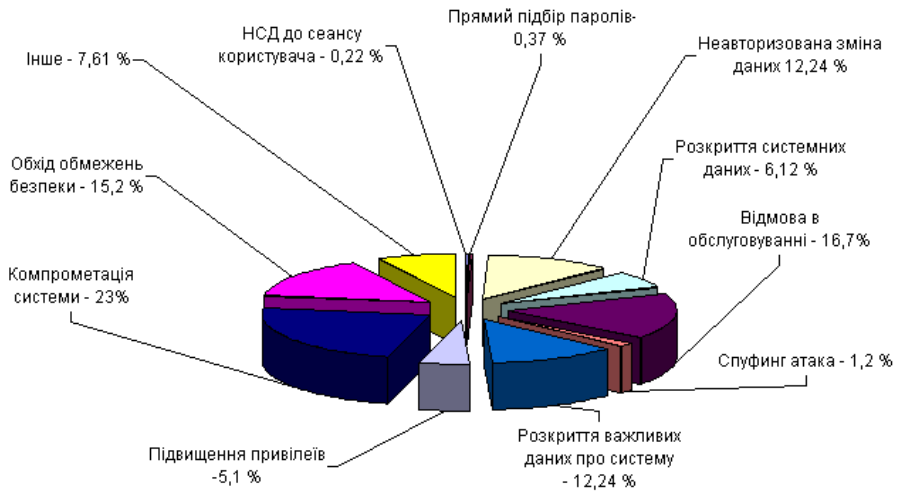


Рис. 2.21. Розподіл частки найбільших загроз для ІС та АС АПК

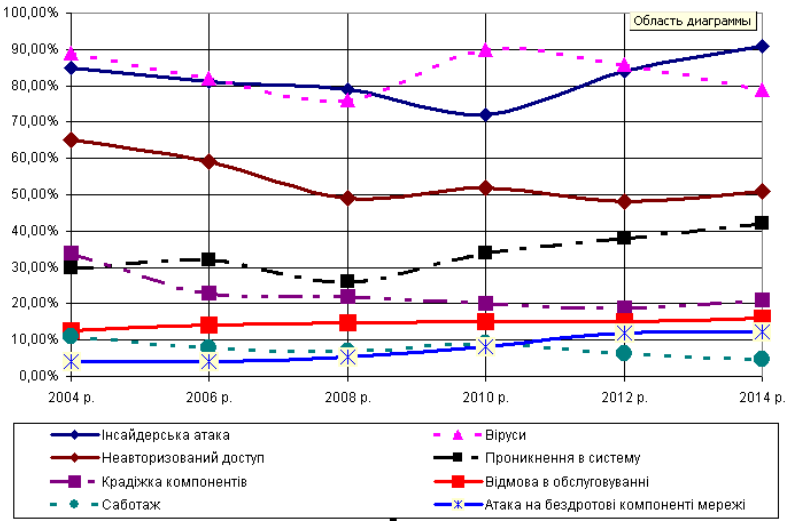


Рис. 2.22. Динаміка найбільших загроз для ІС та АС АПУ
(% за даними респондентів)

Виявлення загроз передбачає комплексний і детальний аналіз усіх факторів, які можуть впливати на безпеку функціонування КВКС. Загрози у відповідності до класів KL (див. табл. 2.4, 2.5, 2.7) поділяються на три базові групи: «потенційні» – дії, які теоретично можуть становити небезпеку; «Реальні» – дії зловмисників по НСД; «Спрямовані» – ті, які спрямовані на реалізацію конкретних уразливостей в КВКС (ІКС та АСК).

Ймовірність реалізації загроз безпеки також знаходиться в одній з груп, які відображають ступінь можливості реалізації загроз безпеці, відповідно до $C_{ППР} = C_{ППР}(\overline{X})$.

Для оцінки ризику реалізації загрози від порушників пропонується використовувати функцію RMI, що характеризує можливість реалізації цієї загрози:

$$RMI = \frac{1}{PAK} \cdot \sum_{i=1}^{PAK} \sum_{j=1}^{PA} MI_{ij}^{no} \cdot PB_j \cdot (kw_1 \cdot LPU_j + kw_2 \cdot \frac{1}{PA} \sum_{i=1}^{PA} U_G^F), \qquad (2.24)$$

$PAK < PA$,

де PAK – кількість небезпечних суб'єктів ІКС;

PA – число можливих цілей порушника в захищеному ІКС;

MI_{ij}^{no} – відомі загроз (потенційно «безпечні») (табл. 2.4);

PB_j – базова ймовірність реалізації загрози, тобто загальновідома або загальноприйнята ймовірність конкретної загрози;

kw_1, kw_2 – вагові коефіцієнти $kw_1 + kw_2 = 1$;

LPU_j – потенційний рівень загрози атаки класу KL_i (табл. 2.4);

U_G^F – коригувальні фактори із специфікації моделі порушника U_G.

Ймовірність вибору зловмисником тієї або іншої загрози (класу загроз) для реалізації визначається розподілом ймовірностей $P(u_{GN_a}) = \{p_{u_{GN_a}}\}$, де $U_G = \{u_{GN_a}\}$ – множина типів зловмисника.

Отже, зважаючи на роботи [1-4 і ін.], матрицю виграшів MG_{Gj}, $J = 1, J$ власника ІКС або АСК при грі з *j-им* типом зловмисника можна представити у такому вигляді:

$$MG_{Gj} = \{vq_G(x_{Gi}, u_{GjN_a})\}, \tag{2.25}$$

де $vq_G(x_{Gi}, u_{GjN_a})$ – виграш власника ІКС, АСК при виборі їм i *-оїї* стратегії й виборі j *-им* типом зловмисника N_a *-оїї* стратегії, у цьому випадку під виграшом розуміємо ризик, що розраховується як [1-7]:

$$vq_G(x_{Gi}, u_{GjN_a}) = P_{ijN_a}^{ug} \cdot LC_{N_a}, \tag{2.26}$$

де $P_{ijN_a}^{ug}$ – ймовірність здійснення N_a *-оїї* загрози j *-им* типом зловмисника при i *-ому* реалізованому проекті СЗІ ІКС за заданий час;

LC_{N_a} – величина втрат від здійснення N_a *-оїї* загрози обумовлена власником ІКС, АСК.

Аналогічно до матриці MG_{Gj}, уведемо в модель параметр $MI_u = \{\omega_j\}$ – множину загроз класу KL_i доступних для реалізації даним типом зловмисника – $U_G = \{u_{N_a}\}$. Ймовірність вибору зловмисником тієї або іншої загрози для реалізації визначається розподілом ймовірностей $P(\omega_j) = \{p_{\omega_j}\}$, у якій, зокрема враховується і його кваліфікація. Для розрахунків $P_{ijN_a}^{ug}$ використовується модель подолання системи захисту (блок модель загроз [10]).

Розгляд ІБ ІКС неможливий без аналізу типу зловмисника, який ставить завдання нападу на інформацію. Ймовірність зіткнення з певним типом зловмисника визначається розподілом ймовірностей $P(u_{Gj}) = \{p_{u_{Gj}}\}$. В основу математичної моделі зловмисника покладена модель Гальтона - Ватсона, побудована на Марковських розгалужених процесах [15].

Відповідно до раніше прийнятих позначень, NA_0, NA_1, NA_2, …, N_{am} – номери загроз на відповідних рівнях захисту ІКС. Тривалість перебування в кожному стані відповідно $\tau_0, \tau_1, \tau_2, ..., \tau_m$.

У даному розділі розглянемо, як приклад, блоки моделі, що становлять парну гру із двома протиборчими сторонами - власник ІКС або АСК і зловмисник.

Загалом гру можна описати такою функцією [12]:

$$\beta_G = (X_G, U_G, Z_G),$$

де $X_G = \{x_{Gi}\}$ – множина стратегій власника ІКС, АСК, тобто можливі проекти побудови СЗІ;

$U_G = \{u_{Gj}\}$ – множина типів зловмисника, тобто деякі стратегії поведінки, притаманні тому або іншому типу зловмисників;

Z_G – функція корисності інформації для власника.

У таблиці 2.9 показані основні результати моделювання парної гри для власника інформації й зловмисника при різних варіантах стратегій сторін, відповідно в межах варіантів інвестування в засоби захисту та дій зловмисників із подолання СЗІ. Тобто, при середньозваженій політиці у сфері інвестування в ЗЗІ значно звужуються показники ймовірностей виграшу зловмисника, навіть для варіанта збільшення кількості стратегій нескінченно.

Тут потрібно зауважити, що в ІКС є досить велика кількість інформаційних ресурсів, які становлять інтерес для зловмисників (див. розділ 1). Для кожного інформаційного ресурсу визначається своє власне дерево (граф) атак. Кожний вузол у цьому графові, відповідно, представляє собою деяку підціль (уразливість), досягнення якої, у разі виконання ряду умов, дозволяє зловмисникові піднятися по дереву на більш високий рівень (до наступної уразливості) і так доти, доки зловмисник не досягне вершини дерева (кінцевої мети). В останньому випадку вважається, що зловмисник успішно реалізував напад на інформацію.

Позначимо через $YZ_{iD_{сзи}}\ PA_i - y$ уразливість на $D_{сзі}$ засобі (рівні) захисту інформації, при цьому вважаємо, що число цілей PA для зловмисників не є випадковою величиною, а навпаки – вона фіксована. Вочевидь, можна припустити, що коли не зазначено інше, то $YZ_{0D_{сзи}} = 1$.

Математичне очікування $\mathrm{M}\hbar\mathrm{YZ}_{1\mathrm{D}_{\text{сзі}}} < 1$. Позначимо через $\mathrm{P}_{\mathrm{YZ}_{\mathrm{iD}_{сзі}}}$ ймовірнісний захід процесу просування зловмисника від одного рівня j СЗІ до іншого j - 1.

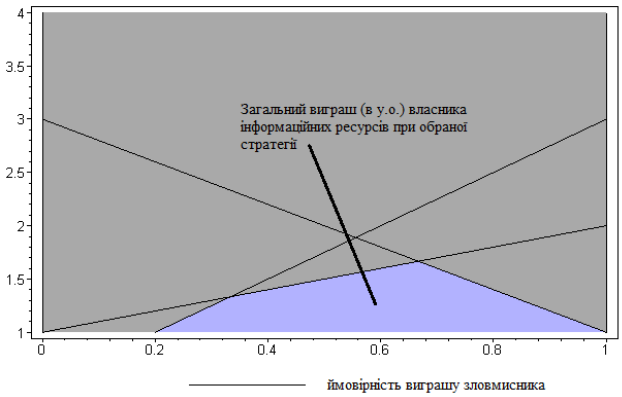
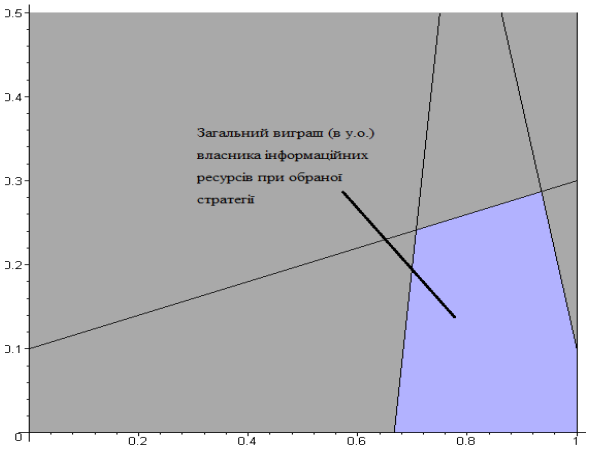
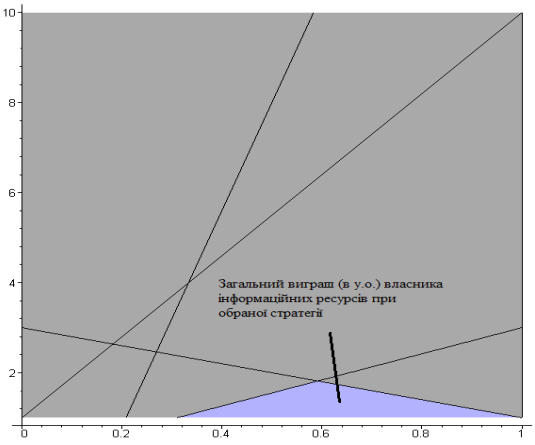
Розподіл ймовірностей $\mathrm{YZ}_{\mathrm{iD}_{\text{сзі}}}$ визначається значеннями:

$$\mathrm{P}_{\mathrm{YZ}_{\mathrm{iD}_{\mathrm{C3I}}}}\{\mathrm{YZ}_{\mathrm{iD}_{\mathrm{DC3I}}} = yk\} = \mathrm{p}_{\mathrm{yk}};\quad yk = 0,1,2,\ldots\quad \sum p_{yk} = 1,$$

де p_{yk} – ймовірність того, що уразливість, яка існує на $D_{сзі}$ засобі (рівні), забезпечує доступ до уразливостей на $(D_{сзі} + 1)$ - рівні.

Таблиця 2.9

Результати моделювання парної гри

У.о. 	Стратегія власника інформації – помірна політика інвестування в СЗІ компанії
У.о. 	Стратегія власника інформації – низький рівень інвестицій у СЗІ компанії
У.о. 	Стратегія власника інформації – середньостатистичний рівень інвестицій у СЗІ компанії

Передбачається, що p_{yk} не залежить від номера рівня СЗІ.

Умовний розподіл $YZ_{iD_{сзі}+1}$ за умови, що $YZ_{iD_{сзі}} = yk$ визначається із припущення, що різні уразливості породжують інші уразливості незалежно

одна від одної. Отже, у цьому випадку величина $\mathrm{YZ}_{\mathrm{iD}_{\text{сзі}}+1}$ розподілена як сума yk незалежних випадкових величин, кожна з яких розподілена також як $\mathrm{YZ}_{1\mathrm{D}_{\text{сзі}}}$. Якщо $\mathrm{YZ}_{\mathrm{iD}_{\text{сзі}}}=0$, то з ймовірністю *1* $\mathrm{YZ}_{\mathrm{iD}_{\text{сзі}}+1}=0$.

Перехідні ймовірності розглянутого Марковського процесу задаються у вигляді:

$$P_{ij}=(\tau,t)=P\{\mathrm{Y}Z_{nk+1}(t)=jk \mid YZ_{nk}(\tau)=D_{i_{\text{сзі}}}\},$$
$$D_{i_{\text{сзі}}},\ jk,\ nk=0,1,2\ldots \qquad (2.27)$$

У процесі дослідження моделі (2.26) використовуються пряме й зворотне рівняння Колмогорова [12] і визначаються розподіл ймовірностей P_{ij} і моменти величини YZ_{nk}. Крім того, моделюється ймовірність того, що випадкова послідовність $\mathrm{YZ}_{\mathrm{iD}_{\text{сзі}}}$ сходить до нуля (зловмисник не може використовувати уразливості для проведення нападу на інформацію), а також поведінка послідовності у випадку, коли вона не сходить до нуля (тобто чи досягне зловмисник мети).

$$\begin{cases}\dfrac{\partial P_{D_{i\text{сзі}}\,\mathrm{y}k}(\tau,t)}{\partial t}=-\mathrm{y}k\cdot\mathrm{p}_{b(t)}\cdot P_{D_{i\text{сзі}}\,\mathrm{y}k}((\tau,t)+\\ +\mathrm{p}_{b(t)})\cdot\sum\limits_{jk=1}^{yk+1}P_{D_{i\text{сзі}},\mathrm{jk}}(\tau,t)\cdot jk\cdot p_{yk-jk+1}(t),\\ P_{D_{i\text{сзі}}\,\mathrm{y}k(\tau,\tau+0)}=\delta_{D_{i\text{сзі}}\,yk},\end{cases} \qquad (2.28)$$

де $\delta_{D_{i\text{сзі}}\,yk}=1$ *при* $D_{i\text{сзі}}=y\mathrm{k}$ і
$\delta_{D_{i\text{сзі}}\,yk}=0$ *при* $D_{i\text{сзі}}\neq y\mathrm{k}$.

$$\begin{cases} \dfrac{\partial P_{D_{i\text{сзі}}\text{у}k}(\tau,t)}{\partial t} = D_{i\text{сзі}} \cdot \text{р}_{b(\tau)} \cdot P_{D_{i\text{сзі}}\text{у}k}((\tau,t) - D_{i\text{сзі}} \cdot \text{р}_{b(\tau)} \cdot P_{D_{i\text{сзі}}\text{у}k}(\tau,t)) - \\ - D_{i\text{сзі}} \cdot b(\tau) \sum\limits_{j\text{п}=D_{i\text{сзі}}=1}^{\infty} P_{D_{i\text{сзі}}\text{у}k}(\tau,t) \cdot p_{j\text{к}-D_{i\text{сзі}}+1}(\tau), \\ \dfrac{\partial P_{0\,D_{i\text{сзі}}\text{у}k}}{\delta\tau} = 0; \quad P_{D_{i\text{сзі}}\text{у}k\,(t-0,t)} = \delta_{D_{i\text{сзі}}\text{у}k} \qquad \textit{при} \quad D_{i\text{сзі}} > 0. \end{cases} \tag{2.29}$$

Вважаємо, що $p_{b(t)\Delta} + p_{o(\Delta)}$ – це ймовірність того, що деяка уразливість, яка в момент часу *t* використовується зловмисником для нападу на інформацію, до моменту часу $(t + \Delta)$ завершиться успіхом. Якщо уразливість використовується в момент τ, то з ймовірностями $p_{D_{i\text{сзи}}}(\tau)$ зловмисникові стають доступні 0, 2, 3, … нових уразливостей. Відповідно до [15], визначаються величини $\text{р}_{\text{b}_{D_{i\text{сзі}}}(t)} = D_{i\text{сзі}} \cdot p_{b(t)}$ й $p_{D_{i\text{сзі}}jk}(t) = p_{jk-D_{i\text{сзі}}+1}(t)$.

У межах нашого дослідження вважаємо, що вплив зловмисника відбуваються у фіксовані моменти часу – t. При вирішенні завдань визначення моментів YZ_{nk} ймовірності того, що зловмисник не зможе використовувати уразливості для проведення нападу на інформацію за заданий час, необхідно враховувати тип графа (дерева) уразливостей.

При оцінці можливості реалізації загроз в моделі, на відміну від багатьох існуючих, враховується фактор часу. У ряді випадків облік цього чинника дозволяє виключити загрози певних класів з числа актуальних.

У дослідженнях [10] показано, що, як правило, використовують три основні типи: двійкове, трійчасте й *ma -e* дерева уразливостей.

Даний алгоритм математичної моделі зловмисника програмно реалізований у середовищі MATLAB 7, Simulink та Delphi і включений у підсистеми імітаційного моделювання (розділ 4).

Результати моделювання дій зловмисника в ІКС наведені в табл. 2.10. Модельована ІКС характеризується чотирма рівнями захисту, двійковим деревом уразливостей $\text{YZ}_{0\text{D}_{\text{сзі}}} - \text{YZ}_{3\text{D}_{\text{сзі}}}$, тривалістю перебування в кожному

стані $\tau_0 - \tau_3$ (задається випадковими числами, рівномірно розподіленими на інтервалі 0 – 20 хв.), приблизний час для проведення атаки $T_{зад}$ = 20 хв.

Таблиця 2.10

Результати моделювання дій зловмисника в ІКС

Ймовірність досягнення мети P за ($T_{зад}$, хв)	Час, витрачений на досягнення мети
P=0,02	$T_{зад}$ = 10 хв
P=0,036	$T_{зад}$ = 14 хв
P=0,08	$T_{зад}$ = 20 хв

Моделювання проводиться до закінчення заздалегідь заданого часу моделювання τ^m, щоб визначити ймовірність P_τ здійснення загрози за заданий час – $P_\tau = NE_{B_{p_a}} / NE$, де $NE_{B_{p_a}}$ – кількість експериментів, у яких за час τ^m був досягнутий стан B_{p_a}; NE – загальна кількість експериментів.

Отримані в результаті моделювання величини P_τ, для різних проектів СЗІ ІКС, АСК, загроз і типів зловмисників, і є шуканими величинами P_{ijk}^{ug}.

Після розрахунків усіх виграшів власника ІКС та АСК у матрицях ігор $\{MG_{Gj}\}$, у кожній матриці розраховується $\{vq_G(x_{Gi}, u_{Gj})\}$ інформаційний ризик за *і -им* проектом СЗІ при зіткненні з *j -м* типом зловмисника. Він розраховується як:

$$vq_G(x_{Gi}, u_{Gj}) = \sum_{yk=1}^{MI} vq_G(x_{Gi}, \omega_{Gjyk}) \cdot p(\omega_{Gjyk}). \qquad (2.30)$$

Для знаходження оптимального проекту СЗІ, тобто стратегії власника ІКС, розглядається матриця Z_G.

Коли відомий розподіл ймовірностей $P(u_{Gj})$, то власник ІКС та АСК може скористатися Байесовською стратегією x_{Gbc} як оптимальним проектом СЗІ, що знаходиться як:

$$Z_G(x_{Gbc}\left|P(u_{Gj})\right)=$$
$$=max(-1)\cdot Z_G(x_{Gi}\left|P(u_{Gj})\right)=max(-1)\sum_j vq_G(x_{Gi},u_{Gj})\cdot p(u_{Gj}). \qquad (2.31)$$

Фактично Байесовська стратегія – це найкраща стратегія власника ІКС або АСК в осередненій грі проти зловмисника. Згідно з [6], якщо Байесовська стратегія збігається з максимінною, то вона є оптимальною. У разі гри з невизначеністю, тобто з невідомим розподілом $P(u_{Gj})$, для знаходження оптимальної стратегії власникові фактично доводиться обирати оптимальну стратегію експертним шляхом, при цьому можна скористатися одним із критеріїв: Вальда, Севіджа, Лапласа або Гурвіца.

2.5. Висновки до розділу 2

У результаті проведених у даному розділі роботи досліджень можна зробити наступні висновки.

1. Визначено, що для проведення ефективної політики ІБ, вибору й впровадження адекватних технічних комплексів СЗІ, необхідно виконати опис, аналіз і моделюванням загроз і уразливостей ІКС та АСК.

2. Показано, що специфіка КВКС до яких відносяться і ІКС та АСК, така, що вони знаходяться під загрозою не тільки проникнення звичайних вірусів, але можуть стати об'єктом спрямованих (цільових) атак з боку зловмисників або кібер-терористів. Серед загроз ІБ, специфічних для ІКС, головними є: використання застарілого ПЗ, обладнання та комунікаційних протоколів; підключення ізольованій мережі ІС та АСК до загальної мережі ІКС або Інтернету. В системах SCADA та HMI, типовими загрозами ІБ є поширене ПЗ для ОС Windows, зокрема, Buffer overflow, DoS/DDoS, SQL Injection, а також схильність PLC-контролерів до мережевих атак на зразок DoS/DDoS.

3. З’ясовано, що складність застосування до систем інтелектуального розпізнавання загроз формалізованого апарату аналізу й синтезу СЗІ ІКС полягає в тому, що конкретний інформаційний комплекс і його підсистема ІБ складаються з різнорідних елементів, які описуються із використанням різних математичних моделей.

4. Показано, що застосування елементів інтелектуального адаптивного захисту ІКС може бути засноване на використанні дискретних процедур розпізнавання загроз ІБ.

5. Розроблено метод інтелектуального розпізнавання загроз (МІРЗ) на основі ДПРЗ із використанням апарату логічних функцій та нечітких множин, що дозволяє підвищити ефективність розпізнавання загроз ІКС залежно від класу до 85–98 %, створювати ефективні аналітичні, схемотехнічні та програмні рішення СЗІ ІКС.

6. Розроблено модель складання вирішального правила для ДПРЗ ІБ, яка дозволяє виконувати інтелектуальне розпізнавання загрози, з мінімальним числом помилок. Показано, що побудова множини елементарних класифікаторів для розглянутих класів загроз, зводиться до знаходження припустимих і максимальних кон’юнкцій для характеристичної функції класу. Описано приклади використання ДПРЗ ІБ для певної мети на основі розрахунків за побудованими елементарними кон’юнкціями.

7. Установлено, що в моделі «голосування» за представницькими наборами при вирішенні завдань інтелектуального розпізнавання загроз ІКС досить обмежитися побудовою представницьких наборів довжини 3. З’ясовано, що ДПРЗ ІБ чутливі до наявності «шуму» у тренувальному наборі. Для подолання цих недоліків як один із варіантів можна використовувати математичний апарат нечітких множин. Визначено, що в завданні інтелектуального розпізнавання КНІ у ІКС, частина значень ознак мають вагу близьку до нуля, але при цьому багато і таких значень, які мають більшу вагу, тобто є дуже типовими для одного із класів загроз ІБ.

8. Запропоновано використовувати показник поточних ризиків реалізації загроз ІКС для оцінки в реальному масштабі часу загроз ІБ.

9. Проаналізовано (із використанням ігрових моделей) ймовірності вибору зловмисником тієї чи іншої загрози для подальшої реалізації. Отримано залежності для моделювання станів системи при реалізації розглянутих класів загроз із використанням напівмарковського процесу.

10. Реалізовано в MATLAB 7 та Simulink модель інтелектуального розпізнавання загроз (ДПРЗ ІБ) із використанням апарату логічних функцій та нечітких множин ознак НСД у ІКС.

11. На основі запропонованого МІРЗ, розроблена експертна система «Аналізатор загроз» для виявлення НСД та складних нападах на інформацію у ІКС, яка працює під операційними системами Windows і не потребує спеціальної підготовки.

12. Обґрунтовано, що для подальшого розв’язання проблеми розвитку моделей інтелектуального розпізнавання загроз ІКС, потрібно зосередитися на розгляді режимів роботи ІКС та АСК у разі втрати заявок внаслідок блокування неоднорідних потоків ДПРЗ, які виникають при складних нападах на інформацію, зокрема DoS/DDoS атаках, у ІКС, або коли втрати з’являються через переповнення заявками у відповідних модулях підсистем клієнт-банк, електронних накладних, e-business, e-logistics, e-cargo, e-ticket, систем зв’язку GSM-R, VSAT, системах SCADA, HMI та ін.

РОЗДІЛ 3
МОДЕЛЮВАННЯ ІНТЕЛЕКТУАЛЬНОГО РОЗПІЗНАВАННЯ ЗАГРОЗ В УМОВАХ АТАК С НЕОДНОРІДНИМИ ПОТОКАМИ ЗАПИТІВ В ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНОМУ СЕРЕДОВИЩІ

Вирішення питань комплексного забезпечення захищеності та стійкості функціонування ІКС в умовах НСД, у тому числі, впливу КНІ у загальнодоступні модулі, вимагає системного аналізу й синтезу можливих варіантів побудови засобів протидії НСД. При формуванні комплексу необхідно узгоджувати та взаємопов'язувати функції й параметри ІКС та АСК, засобів захисту інформації від НСД, антивірусних засобів, міжмережних екранів, комунікаційного устаткування, загального й спеціального програмного забезпечення та перспективних засобів протидії нападам на інформацію.

Методика інтелектуального розпізнавання загроз інформаційно-комунікаційному середовищу та захисту від нападів на інформацію, зокрема, поширених типів комп'ютерних атак (КНІ), повинна включати три компоненти. По-перше, необхідно максимально багато знати про ситуацію, яка передувала атаці. По-друге, виявити і максимально блокувати атаку під час її здійснення. По-третє, проаналізувати, як це все відбувалося, після КНІ. Цього можна домогтися, проаналізувавши багато показників, зокрема, репутацію джерела трафіку, визначені характеристики трафіку, що вказують на можливість атаки, його відхилення від звичайної поведінки і т. п. Для захисту в ході атаки не можна обмежуватися тільки аналізом сигнатур, потрібно використовувати і додаткові методи виявлення, наприклад, фільтри на основі репутації, інтелектуальні моделі розпізнавання, зокрема моделі на основі запропонованих ДПРЗ і т. п.

3.1. Моделі нападів на інформацію в умовах неоднорідних потоків запитів у інформаційно-комунікаційному середовищі

Основні підходи до аналізу уразливостей ІБ ІКС та АСК і оцінки їх рівня захищеності базуються на аналітичних обчисленнях та імітаційному моделюванні. Аналітичні підходи, як правило, використовують різні методи оцінки ризиків [13 і ін.]. Методи імітаційного моделювання ґрунтуються на моделюванні комп'ютерної мережі, деревах атак, моделей графів та ін. [8 і ін.].

Особливістю створюваних ІКС та АСК, наприклад на транспорті, є наявність в них крім центральних серверів опрацювання та зберігання даних на диспетчерському пункті управління транспортної системи, персональних автоматизованих робочих місць (АРМ) безпосередньо на кожному транспортному засобі.

Персональні АРМ технічно являють собою мобільні засоби зв'язку, навігації, міні-комп'ютери, флеш-пристрої, за допомогою яких відбувається організація обміну інформацією з центральним сервером диспетчерського пункту управління транспортної системи через мережу Інтернет з будь-якої точки світу. У загальному випадку таку ІКС можна охарактеризувати як територіально-розподілену телекомунікаційну керуючу систему транспортними об'єктами, а засоби телекомунікації ТЗ – як мобільне АРМ (МАРМ).

Попередній аналіз сучасних тенденцій розвитку ІКС (див. розділ 1) показує, що до їх складу входить велика кількість АРМ на базі мобільних персональних комп'ютерів. Деякі організації вже зараз практично на 100% оснащені технічними засобами, що мають доступ до мережі Інтернет і працюють в ній в режимі «on-line» [10].

Результатом впливів КНІ на МАРМ, наприклад, таких поширених як DoS/DDoS, може бути дестабілізація функціонування ІКС управління технологічним процесом, відкриття існуючих і формування нових каналів доступу, порушення функцій ПЗ, поява даних недостовірного характеру, руйнування ОС, СУБД та ін. Наприклад, хакери КНР в період з 2012 р. по 2013

р. здійснили не менше 15 атак (DoS/DDoS, Probe, R2L) на комп'ютерні мережі транспортних компаній США, що працюють за контрактом на міністерство оборони. При цьому, як мінімум, в дев'яти випадках китайські хакери проникали в АРМ та МАРМ логістичних компаній, завантажували туди шкідливі програми і отримували різноманітні дані, які стосувалися перевезень озброєння та особистого складу [14 та ін.].

Питання про застосування алгоритмів зі зворотним зв'язком у СЗІ ІКС (що враховують наявність і розмір черг заявок, наприклад, запитів у модулях систем клієнт-банк, електронних накладних, e-business, e-logistics, e-cargo, e-ticket, систем зв'язку GSM-R, VSAT та ін., швидкість надходження вимог, зокрема з МАРМ, інтервал між послідовними вимогами, тип вимог, ДПРЗ і т. д.) виникає при більш детальному розгляді так званих циклічних алгоритмів, у яких використовується тільки інформація про вхідні потоки й потоки насичення. Такий режим керування (у якому обслуговування потоків вимог відбувається суворо за заздалегідь визначеним законом) найчастіше застосовується в системах обслуговування з великим завантаженням (наприклад, у системі АСК ВП УЗ-Є Укрзалізниці, щодоби опрацьовується понад 650 тис. запитів. Середній час опрацювання повідомлень – менше 0,8 секунди на кожному з майже 100 потоків опрацювання), коли інтенсивності надходження вимог за різними потоками практично однакові. Проте, у разі появи в потоках розривів (немає заявок, що надійшли), циклічний спосіб керування є недоцільним: для деякого потоку обслуговуюче обладнання працює в неробочому режимі, у те час як за іншими потоками є черги заявок на обслуговування. У таких випадках раціонально застосовувати інші керуючі алгоритми, що використовують додаткову інформацію про структуру вхідних потоків вимог, наприклад при замовленні квитків, плануванні перевезень, бронюванні вантажу та т. п. Однак впровадження подібних алгоритмів вимагає застосування додаткових технічних засобів, а це негайно призводить до здорожчання й ускладнення системи. Отже, виникає питання про розробку більш простих і ефективних алгоритмів зі зворотним зв'язком, що

використовують деяку мінімальну інформацію про систему та не вимагають складного технічного обладнання.

Розглянемо типову структуру ІКС, див. рис. 1.4, 1.7. Обчислювальний комплекс ІКС складається з N_{ser} серверів, на яких працюють додатки та зберігаються бази даних. Інформаційні ресурси ІКС можна поділити на такі типи: текстові ресурси; спеціальне ПЗ; бази даних; графічні ресурси; змішані типи тощо. Кожний тип ресурсів визначає певну політику безпеки та структуру апаратно-програмного комплексу СЗІ. Для розрахунків параметрів СЗІ ІКС та АСК необхідно знати час, який клієнт витрачає під час звернення до ресурсів. Причому на доступ до різних типів ресурсів або обслуговування заявок на цих ресурсах витрачається різний час, що дає RES (RES – число різних ресурсів) розподілів незалежних випадкових величин. Припущення про незалежність цих випадкових величин можна зробити з того, що різні типи ресурсів мають різні алгоритми роботи. Для знаходження загального часу, витраченого для доступу до ресурсів, необхідно знайти розподіл суми незалежних випадкових величин. Як відомо для цього використовується операція згортки (*) RF_i розподілів [10]: $RF_s = RF_1 * RF_2 * ... * RF_n$.

Причому самі розподіли можуть братися як передбачувані теоретичні, так і отримані після серій вимірів. Для моделювання параметрів СЗІ ІКС та АСК можна використовувати положення теорії дискретних Марковських процесів – ланцюгів Маркова [11 і ін.].

Крім того, для моделювання й аналізу загроз для ІБ ІКС використовуються графи атак і мережі Петрі-Маркова [2-14 і ін.].

У межах нашого дослідження будемо вважати, що ІКС та АСК становить собою сукупність локальних вузлів і міжмережних екранів, з'єднаних між собою за допомогою комутаторів відповідно до змішаної топології даних, що використовують для передачі технології Ethernet і стік протоколів TCP/IP.

Для різнорідних заявок в рамках бізнес-процесів, наприклад, на транспорті, зокрема, з інтегрованими системами GSM-R, VSAT, GPS, МАРМ, модулями e-logistics, e-cargo, e-business, e-ticket, та ін., розглянемо випадок

моделі ІКС, обслуговуючого потік заявок від віконних додатків, МАРМ, веб-інтерфейсів користувачів і інших прикладних компонентів.

Позначимо через N_A – множину номерів загроз інформації; $D_{сзі}$ – множину номерів засобів захисту, які можуть бути використані в системі захисту; B_{p_a} – множину номерів загроз інформації, реалізованих порушником при досягненні p_a *-оїмети*; $N_j^{p_a}$ – множину номерів засобів захисту, які потенційно можуть бути використані для протидії реалізації порушником p_a - *оїмети* на j *-му* рубежі захисту (для нейтралізації *j -ї* загрози, що входить в p_a *-у* ціль) (p_a*=1,2,...,PA;* *j=1,2,...,MI*). Причому, $B_{p_a} \subset N_A$, $\bigcup_{p_a=1}^{PA} B_{p_a} = N_A$, $n_{p_a} = \left|B_{p_a}\right|$ і $\bigcup_{p_a=1}^{PA} \bigcup_{j \in B_{p_a}}^{n_{p_a}} N_j^{p_a} \subset D_{сзі}$.

В цьому випадку процес реалізації порушником кожної зі своїх цілей може бути представлений у вигляді спрямованого графа, приклад якого наведено на рис. 3.1. Вершини графа представляють собою стани ІКС, що відповідають спробам реалізації порушником деякої загрози інформації. Стан системи S_0 є початковим, тобто таким, при якому ще жодна із загроз інформації не реалізована. Стан *Sj* ($j \in B_{p_a}$) відповідає спробі реалізації *j-ї* загрози. У разі її успішної реалізації, здійснюється перехід до наступного стану системи (відповідна дуга графа), а якщо ні, то (при штатному реагуванні СЗІ) здійснюється перехід до стану $S_{n_{pa}+1}$ (на рис. 3.1 $S_{n_{pa}+1} \equiv S_8$). Стан $S_{n_{pa}}$ є кінцевим і відповідає досягненню порушником p_a *-ї* мети (p_a*=1,2,..., PA*). Кожна дуга характеризується значенням ймовірності переходу між станами системи. Пунктиром позначені дуги, що відповідають переходу з даного стану в стан $S_{n_{pa}+1}$.

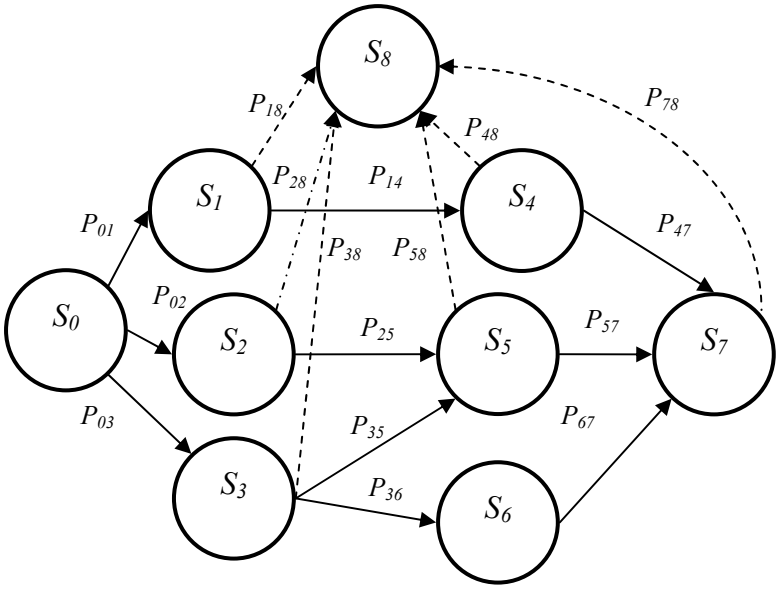


Рис. 3.1. Граф станів ІКС

Ймовірність знаходження системи в *k -му* стані, при спробі реалізації порушником p_a *-ї* мети, буде визначатися таким виразом:

$$P_k^{p_a} = \sum_{l \in G_{i-1}^{p_a}} P_l^{p_a} p_{lk}^{p_a}, \quad k \in G_i^{p_a}, \; i = 0,1,2,..., I^{p_a} \;\; p_a = 1,2,..., PA,$$

де I^{p_a} – число рівнів у ранжируваному графові станів, що описує діяльність порушника при спробі досягти p_a *-ї* мети;

$G_i^{p_a}$ – множина номерів вершин, що становлять *i -й* рівень графа станів, що описує діяльність порушника при спробі досягти p_a *-ї* мети, причому:

$$\bigcup_{i=0}^{I^{p_a}} G_i^{p_a} \subset B_{p_a}; \; p_{lk}^{p_a} = \rho_{lk}^{p_a} g_l^{p_a};$$

$g_j^{p_a}$ – ймовірність подолання *j -го* рубежу захисту при спробі досягнення порушником p_a *-ї* мети $g_j^{p_a} = \left(1 - e^{-Kq_j K \omega_{p_a}}\right) \prod_{m \in N_j^{p_a}} \left(1 - r_{jm}^{p_a} x_{jm}\right)$;

$r_{jm}^{p_a}$ – ймовірність успішного функціонування *m-го* засобу захисту із протидії діяльності порушника на *j-му* рубежі при спробі досягти ним p_a *-ї* мети ($j \in B_{p_a}$; p_a*=1, 2,..., PA;* $m \in N_j^{p_a}$);

K_{q_J} – коефіцієнт узгодження при переході системи в *j -ий* стан;

$K_{\omega_{p_a}}$ – рівень кваліфікації порушника при спробі досягнення p_a *-ї* мети, $K_{\omega_{p_a}} \in [0,1]$, при спробі реалізації p_a *-ї* мети (p_a *=1,2,...,PA*), $x_{jm} = \{0,1\}$;

x_{jm}*=1*, якщо *m-ий* засіб використовується на *j -му* рубежі захисту, x_{jm}*=0* – якщо ні, то ($j \in B_{p_a}, j \neq 0, j \neq MI+1$; $m \in N_j^{p_a}$);

$\rho_{lk}^{p_a}$ – ймовірність переходу з *l* -го стану графа в *k -ий* при спробі реалізації порушником p_a *-ї* мети.

У даній частині роботи розглянутий алгоритм, при якому апріорі виділяються найбільш інтенсивні вхідні потоки ІКС, потоки найбільш важливі в сенсі оперативності обслуговування при управлінні процесами у ІС АПК й потоки малої інтенсивності. У процесі обслуговування такий алгоритм ураховує наявність черг за деякими потоками, що вимагають швидкого обслуговування СЗІ у складі ІКС.

Множинний доступ до ресурсів ІКС передбачає поділ ресурсів між абонентами модулів e-business, систем відеоспостереження, МАРМ, каналів систем зв’язку GSM-R, VSAT та ін.

Зловмисники потенційно здатні, організовувати КНІ DoS/DDoS на комунікаційні системи залізничних ліній, що може призвести до виходу з ладу останніх на тривалий термін [1]. Елементи подібних систем існують ізольовано від інтернету, обмінюючись даними по стандарту GSM-R (стандарт стільниковий зв'язку з додатковим криптографічним захистом сигналу). Однак, той факт, що програмні ключі, які використовуються для створення зашифрованого з'єднання в мережах комунікацій залізничних колій, зазвичай зберігаються на фізичних носіях і пересилаються по відкритих каналах зв'язку,

може означати, що одного разу подібний ключ виявиться в руках зловмисників, або навіть терористів, і ризик атаки серйозно зросте (розділ 2).

Аналогічна ситуація має місце і у системах VSAT (мережі VSAT будуються на базі геостаціонарних супутників-ретрансляторів), які використовуються на морському та трубопровідному транспорті для зв'язку з віддаленими об'єктами. Наприклад, у 2002-2003 були зафіксовані спроби здійснення КПІ DoS/DDoS на окремі морські судна [2].

Назвемо потоки неоднорідними (конфліктними), якщо, по-перше, неможливо підсумувати деякі потоки й звести завдання до одномірного випадку, по-друге, обслуговування заявок неоднорідних потоків здійснюється в інтервали часу, що не перетинаються, по-третє, існують інтервали неприступності, протягом яких потоки не обслуговуються внаслідок використання апарату ДПРЗ ІБ.

Розглянемо багатолінійну СМО (ІКС) з накопичувачем кінцевої ємності. У системі існує n_{pr} працюючих незалежно друг від друга ідентичних приладів, які обслуговують однотипні заявки, що надходять на них. Кожне з обладнань ІКС та АСК із СЗІ може перебувати на одній з $T_{fo}, 1 < T_{fo} < \infty$ фаз обслуговування. Час обслуговування заявки на кожному обладнанні розподіллено за законом фазового типу з параметрами $h_в$ і H, де $h_в$ – вектор-рядок розмірності T_{fo}, а H – квадратна матриця порядку T_{fo}. Функція розподілу фазового типу часу обслуговування заявки записується у вигляді – $H(x) = 1 - h_в e^{H\tau} 1$, де 1 – вектор-стовпець із одиниць, (далі будемо позначати через *0* – нульову матрицю, а через *ES* – одиничну матрицю).

Якщо в ІКС або АСК надходить заявка, але все n_{pr} обладнання зайняте, то ця заявка потрапляє у накопичувач (чергу) ємністю NO_i. Як правило, $NO > 2$; якщо накопичувач NO_i повний, то заявка залишає систему без обслуговування (губиться). Заявки з накопичувача обслуговуються в порядку їх надходження до системи. Крім цього, позначимо $R = n_{pr} + NO$.

Розглянемо процес із кінцевою множиною станів $\{1, 2,..., I_s\}$, $1 \le I_S < \infty$. У кожний момент зміни стану процесу генерується нова заявка, яка готова надійти в ІКС на обслуговування. Ймовірність того, що процес за час менше τ перейде зі стану i відразу в стан $j, i, j = \overline{1, J}$ дорівнює $Wij_{(\tau)}$. Середній час між змінами станів ІКС в стаціонарному режимі можна записати у вигляді:

$$\tau_s = \pi_{ma} \int_0^\infty \tau \cdot dW(\tau)1,$$

де π_{ma} – вектор-рядок стаціонарних ймовірностей вкладеного ланцюга Маркова;

$W(\tau)$ – матриця з елементів $Wij_{(\tau)}$.

Розглянемо стаціонарний режим функціонування процесу генерації заявок на обслуговування у ІКС. За час $\Delta\tau$ із ймовірністю p_α відбувається блокування потоку заявок, що надходять у систему, а саме, починаючи із цього моменту, заявки, що генеруються процесом, у систему не потрапляють, а «губляться». Заявки, що перебувають у системі, систему не залишають, а продовжують обслуговуватися (заявки на обладнаннях СЗІ) або очікувати обслуговування (заявки в черзі). Якщо потік блокований, то за час $\Delta\tau$ із ймовірністю p_β потік розблокується, і заявки, які будуть згенеровані після цього моменту, знову надходитимуть у систему на обслуговування. Генерація заявок потоком не залежить від того, блоковане надходження заявок у систему чи ні.

Скористаємося деякими побудовами, отриманими для багатолінійної СМО з кінцевим накопичувачем, а саме тим, що процес обслуговування всіма обладнаннями СЗІ, обслуговування на кожному з яких розподілено за законом фазового типу, може бути описаний у вигляді Марковського процесу обслуговування.

Якщо в системі перебуває κ_z, $N0<\kappa<R$, заявок, то процес обслуговування може перебувати в одному з $T_{fo}^k, T_{fo}^k < \infty$ станів (фаз обслуговування), причому

інтенсивність зміни фаз Марковського процесу визначається елементами матриць L_k, $k_z = \overline{0, n_{pr} + NO}$, якщо жодна заявка не була обслужена, і елементами матриць M_k, $k_z = \overline{1, n_{pr} + NO}$, якщо заявка була обслужена. Припускається, що $T_{fo}^k = T_{fo}$ при $k_z = \overline{n_{pr}, n_{pr} + NO}$ матриці $L_k = L$ збігаються при $k_z = \overline{n_{pr}, n_{pr} + NO}$, а матриці $M_k = M$ збігаються при $k_z = \overline{n_{pr} + 1, n_{pr} + NO}$. Матрицю $L+M$ будемо вважати нерозкладною, а матрицю M - ненульовою.

Якщо в системі перебуває k_z, $k_z = \overline{0, n_{pr} - 1}$ заявок, то припустимо, що в момент надходження чергової заявки в систему те, на яку фазу перейде Марковський процес обслуговування, буде визначатися елементами матриць Ω_k.

Розглянемо вкладений ланцюг Маркова, обумовлений моментами зміни фаз процесу генерації заявок.

Позначимо через $p^w{}_{ik}$, $w = 0,1$, $i = T_{fo}^k \cdot (u_f - 1) + v_f$, $u_f = \overline{1, I_s}$, $u_f = \overline{1, I_s}$, $v_f = \overline{1, T_{fo}^k}$, $k_z = \overline{0, R}$ стаціонарну ймовірність того, що відразу після зміни фаз процесу в системі перебуває k_z заявок, фаза напівмарковського процесу генерації заявок перебуває на фазі u_f, Марковский процес обслуговування перебуває на фазі v_f, отже, якщо $w = 0$, то потік заявок заблокований, а, якщо $w = 1$, то потік заявок розблокований. Припустимо, $p_{k_z}^w = (p_{1k_z}^w, ..., p_{1z_k, k_z}^w)$, $p_{k_z} = (p_{k_z}^0, p_{k_z}^1)$, $w = 0,1$, $k_z = \overline{0, n_{pr} + NO}$.

Для вектора $\boldsymbol{p}$ слушна система рівнянь рівноваги (СУР) $\boldsymbol{p} = \boldsymbol{p} \cdot G_M$, у якій матриця G_M є матрицею перехідних ймовірностей вкладеного ланцюга Маркова. Матрицю G_M можна представити в блоковому вигляді:

$$G_M = \begin{pmatrix} G_{00} & G_{01} & 0 & 0 \ldots & 0 & 0 \\ G_{10} & G_{11} & G_{12} & 0 \ldots & 0 & 0 \\ \ldots\ldots & & & & & \\ G_{R-1,0} & G_{R-1,1} & G_{R-1,2} & G_{R-1,3} \ \ldots & G_{R-1,R-1} & G_{R-1,R} \\ G_{R0} & G_{R1} & G_{R2} & G_{R3} \ \ldots & G_{R,R-1} & G_{R,R} \end{pmatrix}.$$

Методи розв'язання систем рівнянь рівноваги описані в роботах [1]. Для нашої моделі введемо для матриць G_{Mij} деякі додаткові позначення. Позначимо через $q_{ij}(\tau),\quad i,j=0,1$ ймовірність того, що через час τ заявки не будуть надходити в систему на обслуговування (надходження заявок буде заблоковано). Якщо $j=0,$ і заявки будуть надходити в систему, то надходження заявок буде розблоковане. Якщо $j=1$ – надходження заявок заблоковане. Тоді доречні такі співвідношення: $q_{00}(\tau)+q_{01}(\tau)=1$ та $q_{10}(\tau)+q_{11}(\tau)=1.$

Знайдемо ймовірність $q_{ij}(\tau),\quad i,j=0,1$. Перетворення функції Лапласа $q_{00}(\tau)$ позначимо як $\hat{q}_{00}(s)$, тоді, $\hat{q}_{00}(s)=\int_0^\infty e^{-s\cdot\tau}q_{00}(\tau)d\tau.$

Припустимо, що в початковий момент часу надходження заявок заблоковано. Тоді через час τ надходження заявок залишиться заблокованим, якщо за час τ стан блокування взагалі не змінювався, або змінювався парне число раз. Тому для $\hat{q}_{00}(s)$ в силу незалежності періодів часу, протягом яких процес надходження заявок у систему перебуває в заблокованому та розблокованому станах, доречне співвідношення:

$$\hat{q}_{00}(s)=\frac{1}{s+p_\beta}+\frac{p_\alpha\cdot p_\beta}{(s+p_\beta)^2\cdot(s+p_\alpha)}+\frac{p_\alpha^{\,2}\cdot p_\beta^{\,2}}{(s+p_\beta)^3\cdot(s+p_\alpha)^2}+\ldots=\frac{s+p_\alpha}{s\cdot(s+p_\alpha+p_\beta)}.$$

Отже, ймовірність $q_{00}(\tau)$ можна знайти зі співвідношення

$$q_{00}(\tau)=\frac{p_\alpha}{p_\alpha+p_\beta}+\frac{p_\beta}{p_\alpha+p_\beta}\cdot e^{-(p_\alpha+p_\beta)\cdot\tau}.$$

Користуючись аналогічним підходом можна одержати вираз для $q_{01}(\tau),\ q_{10}(\tau),q_{11}(\tau)$, тобто:

$$q_{01}(\tau) = \frac{p_\beta}{p_\alpha + p_\beta} - \frac{p_\beta}{p_\alpha + p_\beta} \cdot e^{-(p_\alpha + p_\beta)\cdot\tau},$$

$$q_{10}(\tau) = \frac{p_\alpha}{p_\alpha + p_\beta} - \frac{p_\alpha}{p_\alpha + p_\beta} \cdot e^{-(p_\alpha + p_\beta)\cdot\tau},$$

$$q_{11}(\tau) = \frac{p_\beta}{p_\alpha + p_\beta} + \frac{p_\alpha}{p_\alpha + p_\beta} \cdot e^{-(p_\alpha \cdot p_\beta)\cdot\tau}.$$

Матриці перехідних ймовірностей $G_{M\,ij}$, з урахуванням результатів досліджень, викладених у роботах [3], можна подати в такому вигляді:

$$G_{M\,i,i+1} = \begin{pmatrix} 0 & W_{i,i}^{01} \\ 0 & W_{i,i}^{11} \end{pmatrix}, \qquad i = \overline{0, n_{pr} - 1},$$

$$G_{M\,i,i+1} = \begin{pmatrix} 0 & W_{0}^{01} \\ 0 & W_{0}^{11} \end{pmatrix}, \qquad i = \overline{0, R - 1},$$

$$G_{M\,R,R} = \begin{pmatrix} W_{0}^{00} & W_{1}^{01} + W_{0}^{01} \\ W_{0}^{10} & W_{1}^{11} + W_{0}^{11} \end{pmatrix},$$

$$G_{M\,i,0} = \begin{pmatrix} W_{i,0}^{00} & 0 \\ W_{i,0}^{10} & 0 \end{pmatrix}, \qquad i = \overline{0, R}, \qquad (3.1)$$

$$G_{M\,i,j} = \begin{pmatrix} W_{i,j}^{00} & W_{i,j-1}^{01} \\ W_{i,j}^{10} & W_{i,j-1}^{11} \end{pmatrix}, \qquad i = \overline{0, R}, \quad j = \overline{1, \min\{i, n_{pr} - 1\}},$$

$$G_{M\,i,n} = \begin{pmatrix} W_{i-n}^{00} & W_{i,n-1}^{01} \\ H_{i-n}^{10} & W_{i,n-1}^{11} \end{pmatrix}, \qquad i = \overline{n_{pr}, R},$$

$$G_{i,j} = \begin{pmatrix} W_{i-j}^{00} & W_{i-j+1}^{01} \\ W_{i-j}^{10} & W_{i-j+1}^{11} \end{pmatrix}, \qquad i = \overline{n_{pr} + 1, R}, \quad j = \overline{n_{pr} + 1, \min\{i, R - 1\}}.$$

Особливістю функціонування даної ІКС із відмовами є те, що при однакових параметрах процесу та параметрах часу обслуговування в системі з переривчастим потоком навантаження на обладнання менше в порівнянні зі звичайною системою. Знайдемо стаціонарні ймовірності χ_0 й χ_1 того, що в момент генерації довільної заявки надходження заявок було заблоковано та розблоковано відповідно. Середній час періоду блокування надходження заявок

дорівнює $1/P_\beta$, а середній час періоду розблокування процесу надходження заявок – $1/P_\alpha$. У цьому випадку шукані ймовірності можна розрахувати з таких виразів:

$$\chi_0 = \frac{P_\alpha}{P_\alpha + P_\beta} \text{ та } \chi_1 = \frac{P_\beta}{P_\alpha + P_\beta}.$$

Таким чином, отримуємо співвідношення для розрахунків наступних стаціонарних характеристик:

розподіл числа заявок у системі в моменти зміни станів процесу й за часом, розподіл часу очікування обслуговування й часу перебування прийнятої до обслуговування заявки;

частка часу простою окремого обладнання СЗІ.

Відповідні залежності, отримані в ході моделювання, показані на рис. 3.2.

Більшість робіт, що стосуються режимів функціонування інфокомунікаційних систем, ґрунтуються на припущенні, що тривалості інтервалів між послідовними надходженнями запитів в ІКС та АСК розподілені за показовим законом. Це дозволяє представляти вхідні потоки вимог (запитів) потоками Пуассона.

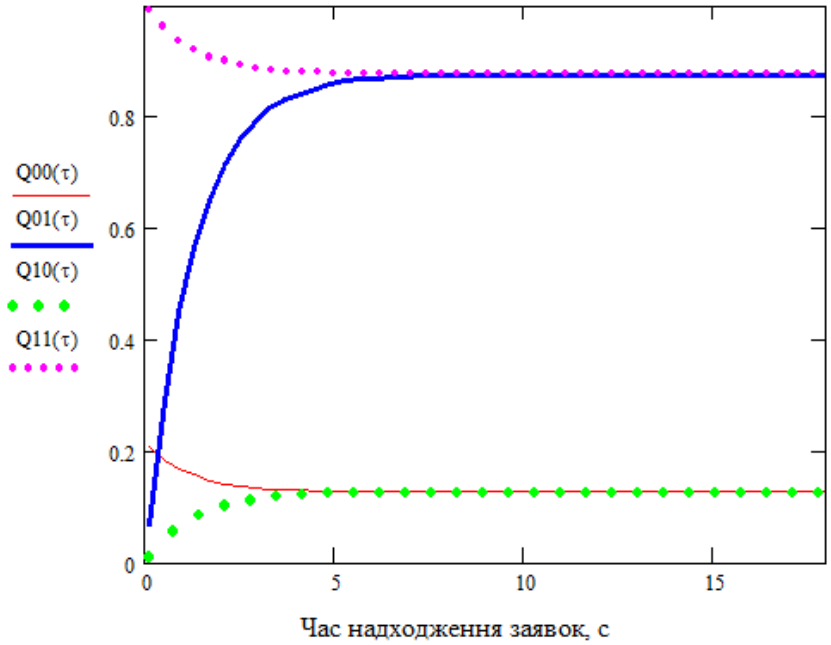


Рис. 3.2. Результати моделювання ймовірностей $q_{ij}(\tau)$

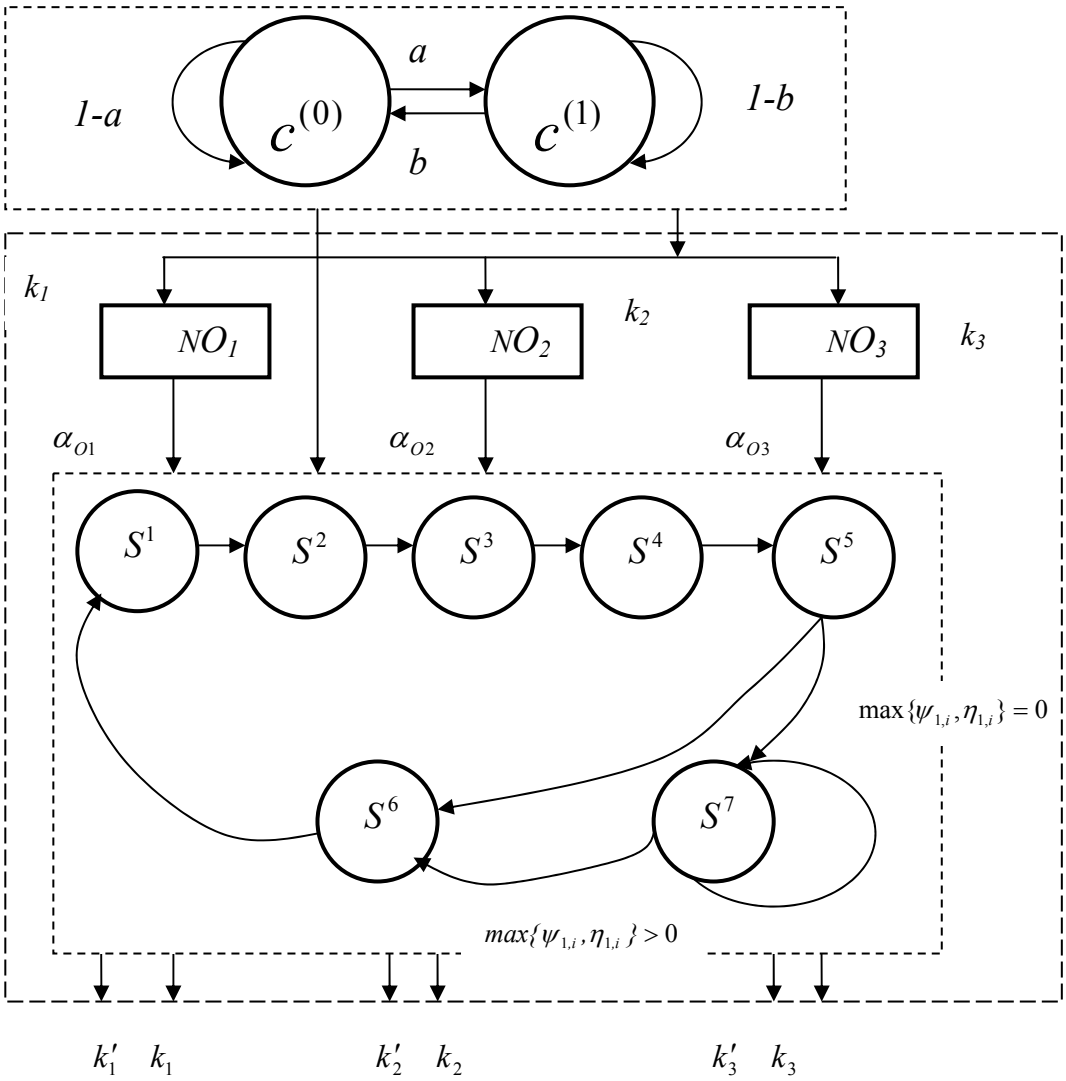
Однак при певних ситуаціях, наприклад, коли зловмисник ставить завдання виводу ІКС із нормального режиму роботи, не можна говорити про незалежність надходження запитів на опрацювання. У цьому випадку потоки запитів не є потоками Пуассона. Для потоків такої структури адекватною математичною моделлю є потік Бартлетта [3].

На сьогодні залишається маловивченим питання моделювання ІС із змінною структурою, у тому числі СЗІ, що представляють собою математичні моделі поведінки об'єктів із вхідними потоками вимог в умовах їх неоднорідності (конфліктності).

Спираючись на вищенаведений аналіз можливості моделювання КНІ у ІКС та АСК і відповідно, СЗІ, функціональну схему КНІ для системи можна представити так (див. рис. 3.3).

Вхідні потоки k_1, k_2, k_3 формуються в деякому випадковому середовищі (ВС), стан якого визначає ймовірнісну структуру цих потоків. Якщо середовище перебуває в стані $c^{(0)}$, то вхідні потоки представляють собою потоки типу Пуассона (потоки окремих вимог). При стані середовища $c^{(1)}$ вхідні потоки є потоками типу Бартлетта (потоки пачок) [7]. Зміна стану середовища визначається показником поточних ризиків $C_{ПТР}$ реалізації загрози певного класу KL_i, наприклад, КНІ Probe, R2L, U2R або DoS/DDoS для ІКС.

Заявки вхідних потоків, наприклад з МАРМ, АРМ та ін., надходять у накопичувачі (черги) $N\text{O}_1, N\text{O}_2, N\text{O}_3$ з необмеженими ємностями. Далі будемо вважати: k_1 – малоінтенсивний пріоритетний потік; k_2 – малоінтенсивний потік; k_3 – пріоритетний потік найбільшої інтенсивності. Інформативність потоку k_1 означає, що в динаміці роботи системи враховується наявність заявок у накопичувачі $N\text{O}_1$ та надходження вимог по цьому потоку. Його пріоритетність – необхідність оперативного обслуговування вимог, що надійшли. Пріоритетність потоку k_3 означає, що при відсутності вимог по потоку k_1 (розрив) буде продовжене обслуговування по потоку k_3.



S^1 – початковий стан входу в ІКС; S^2 – початок сканування доступних ресурсів; S^3 - очікування відповіді про наявність вільних ресурсів; S^4 - підключення до наявних ресурсів; S^5 - передача даних у КС (наприклад, завантаження експлойта); S^6 - передача даних на доступні АРМ або МАРМ; S^7 - завантаження відправлення запитів на сервери.

Рис. 3.3. Функціональна схема КНІ з неоднорідними потоками запитів

Згідно з цими міркуваннями організована робота обслуговуючого обладнання (ОО, наприклад, серверів ІКС, АРМ та ін.), що має стан $S^{(r)}, r = \overline{1,7}$, який утворює множину $S = \left\{S^{(r)} : r = \overline{1,7}\right\}$. ОО в стані $S^{(r)}$ перебуває в перебігу часу $\tau_r, r = \overline{1,7}$. Обслуговуюче обладнання виконує функції з аналізу й обслуговування вимог, із керування вхідними потоками, формування черг у накопичувачах і з відбору вимог із черг за допомогою деяких стратегій обслуговування $\alpha_{O1}, \alpha_{O2}, \alpha_{O3}$. Стан $S^{(2j-1)}$ для $j = 1,2,3$

обслуговуючого обладнання відповідає обслуговуванню вимог потоку k_j. У стані $S^{(2j)}$ для $j = 1,2,3$ не обслуговуються вимоги жодного з вхідних потоків. У стані $S^{(7)}$ обслуговуються вимоги потоку k_3. Згідно із графом, при кожному $r = 1,2,3,4$ стан $S^{(r)}$ переходить у стан $S^{(r+1)}$.

Вихідні потоки при роботі системи з максимальним завантаженням, коли по будь-якому потоку k_i зловмисники, що атакують систему, можуть створити чергу, а ОО працює без простоїв, назвемо потоками насичення й позначимо як k_1', k_2', k_3'. Реальні вихідні потоки в системі будемо позначати k_1, k_2, k_3.

У межах дослідження ми не розглядаємо всі варіанти організації зловмисниками різних неоднорідних (конфліктних) типів потоків запитів, тому що даному питанню присвячені окремі дослідження [10]. Пригадаємо лише невелику кількість варіантів, що підтвердили свою високу ефективність при використанні зловмисниками, зокрема, при КНІ у ІКС [11]: низькошвидкісні КНІ; КНІ з посилкою пакетів з нульовою частотою щодо часової шкали часу проходження пакетів по каналу зв'язку до адресата й назад; КНІ, у яких зловмисник може варіювати тривалість імпульсів; КНІ з мінімальними випадковими значеннями щодо часової шкали часу проходження пакетів по каналу зв'язку до адресата й назад; ін.

Усі аналізовані далі випадкові об'єкти, застосовувані при побудові математичної моделі та пов'язані із процесом обслуговування заявок, будемо задавати на деякому повному ймовірнісному просторі $(\Omega, A, P(*))$ елементарних випадкових подій $\omega \in \Omega$ з ймовірнісним заходом $P(A)$. Для опису вхідних потоків заявок будемо використовувати нелокальний спосіб. Тобто, нашому розгляду підлягає не конкретна вимога, а весь потік заявок.

Довільний вхідний потік k_j описується векторною випадковою послідовністю $\{(\tau_i, \nu_i, \eta_{j,i}), i \geq 0\}$, де $\eta_{j,i}$ – число заявок типу, що ν_i надійшли за проміжок часу $[\tau_i, \tau_{i+1})$ по цьому потоку. Тип заявок визначений міткою

ν_i (станом випадкового середовища). Поведінку випадкового середовища, для простоти, будемо описувати однорідною Марковською послідовністю $\{\nu_i; i \geq 0\}$ із двома станами $c^{(0)}$ – потік заявок з малою інтенсивністю, $c^{(1)}$ – великий потік заявок та ймовірностями переходу $a, b \quad 0 \leq a < b << 1$. Такі обмеження означають, що зміна інтенсивності потоку відбувається рідко, отже, звичайний режим роботи ІКС та АСК із малоінтенсивним потоком заявок буває частіше, ніж потік з великою кількістю запитів, як це відбувається при складних КНІ. Подібні висновки дозволяють вважати, що за час τ_r, коли ОО перебуває в стані $S^{(r)}$, інтенсивність запитів не змінюється. Відомо, що випадкові елементи $\nu_i; i \geq 0$ пов'язані співвідношеннями:

$$\nu_{i+1} = \phi_i(\nu_i, \omega_i), \tag{3.2}$$

де ϕ_i – деякі вимірні відображення простору $\{c^{(0)}, c^{(1)}\} \cdot \{0,1\}$ на $\{c^{(0)}, c^{(1)}\}$;

$\{\omega_i; i \geq 0\}$ – послідовність незалежних випадкових величин з деяким розподілом, у нашому випадку, рівномірним на інтервалі $(0,1)$.

Процеси обслуговування, що протікають, мають у нашій моделі дискретний характер і розглядаються на інтервалах часу, породжуваних деяким випадковим точковим $\tau = \{\tau_i; i \geq 0\}$ процесом на осі часу. Моменти $\tau_i; i \geq 0$, як правило, певним чином пов'язані з моментами зміни станів обслуговуючого обладнання, їх визначення буде подано нижче. Позначимо через $\psi_{j,i}$ довжину черги в накопичувачі NO_j по потоку k_i в момент $\tau_i; i \geq 0$, $j = 1,2,3$. У будь-який момент часу $\tau > 0$ ОО перебуває в деякому стані $S(\tau) \in S$. Керування вхідними потоками й трансформаціями станів ОО з урахуванням вищевказаних попередніх зауважень можна описати так:

$$S_{i+1} = u\left(S_i, \psi_{1,i}, \eta_{1,i}\right) = \begin{cases} S^{(1)} & при\ S_i = S^{(6)}; \\ S^{(r+1)} & при\ S_i = S^{(r)}\ r = \overline{1,4}; \\ S^{(6)} & при\ S_i \in \left\{S^{(5)}, S^{(7)}\right\} \& \max\left\{\psi_{1,i}, \eta_{1,i}\right\} > 0; \\ S^{(7)} & при\ S_i \in \left\{S^{(5)}, S^{(7)}\right\} \& \max\left\{\psi_{1,i}, \eta_{1,i}\right\} = 0; \end{cases} \quad (3.3)$$

для $i = 0,1,\ldots k.$

Для станів ОО припускаємо, що $S_i = S(\tau_i) = S(\tau_i + 0), S(t) = S(\tau_i)$. Випадковий точковий процес $\{\tau_i; i \geq 0\}$ при $\tau_0 = 0$ визначається рекурентним співвідношенням:

$$\tau_{i+1} = \tau_i + U(S_i),\ i \geq 0\ , \quad (3.4)$$

де $U(*)$ – відображення множини S на числову множину $\{T_1, T_2, \ldots T_7\}$ таке, що $T_r = U\left(S^{(r)}\right) > 0, \quad r = 1,2,\ldots,7$.

Будемо називати T_r тривалістю фази (стану) $S^{(r)}$ обслуговуючого обладнання, а величину $T = \sum_{r=1}^{7} T_r$ – тривалістю періоду ОО.

Позначимо через $\zeta_{j,i}, j = 1,2,3..n$ максимально можливе число обслужених на інтервалі часу $[\tau_i, \tau_{i+1})$ вимог потоку k_j при наявності в накопичувачі NO_j нескінченної черги. Тоді відповідний потік насичення k'_j може бути описаний за допомогою точкового процесу – $\{(\tau_i, \nu'_i, \zeta_{j,i}), i \geq 0\}$, де $\nu'_i = (S_i, \nu_i)$ мітка обслужених заявок у ІКС на інтервалі $[\tau_i, \tau_{i+1})$. Інтерпретувати подібний опис ν'_i можна як вплив випадкового середовища на механізм обслуговування.

Більш докладно цей процес буде розглянуто далі в роботі. Ми не будемо задавати розподілу точкових процесів $\{(\tau_i, \nu_i, \eta_{j,i}), i \geq 0\}$ і $\{(\tau_i, \nu'_i, \zeta_{j,i}), i \geq 0\}$ оскільки при нелокальному описі вхідних потоків і потоків насичення можна

обмежитися деякими властивостями умовних розподілів дискретних компонентів $\{\eta_{j,i}; i \geq 0\}$ і $\{\zeta_{j,i}; i \geq 0\}$.

Припустимо, що величина $\overline{\zeta}_{j,i}$, $j = 1,2,3$ задає на проміжку $[\tau_i, \tau_{i+1})$ число фактично обслужених заявок потоку k_j. Для опису реального процесу обслуговування потрібно при будь-якому $i \geq 0$ й кожному $j = 1,2,3$ вказати залежність

$$\overline{\zeta}_{j,i} = f_{j,i}(\psi_{j,i}, \eta_{j,i}, \zeta_{j,i}), \tag{3.5}$$

тобто деяку стратегію α_{O0} ОО. На вибір функції (3.5) природно накласти такі обмеження:

$$0 \leq f_{j,i}(\psi_{j,i}, \eta_{j,i}, \zeta_{j,i}) \leq \zeta_{j,i} \quad \text{та} \quad 0 \leq f_{j,i}(\psi_{j,i}, \eta_{j,i}, \zeta_{j,i}) \leq \psi_{j,i} + \eta_{j,i}.$$

Звідки отримаємо:

$$0 \leq f_{j,i}(\psi_{j,i}, \eta_{j,i}, \zeta_{j,i}) \leq \min\{\psi_{j,i} + \eta_{j,i}, \zeta_{ij}\}. \tag{3.6}$$

Система, як правило, за проміжок часу $[\tau_i, \tau_{i+1})$ обслуговує максимально можливе число заявок $\zeta_{j,i}$ з потоку k_j або всі ті заявки, що надійшли та перебувають у черзі цього потоку, якщо їх число менше $\zeta_{j,i}$.

Тоді залежність (3.5) буде мати вигляд:

$$\overline{\zeta}_{j,i} = \min\{\psi_{j,i} + \eta_{j,i}, \zeta_{j,i}\}. \tag{3.7}$$

Така стратегія механізму обслуговування, враховуючи (3.6), називається екстремальною.

Будемо описувати поведінку системи маркованим точковим процесом $\Lambda = \{(\tau_i, S_i, \nu_i, \psi_i); i \geq 0\}$ з виділеним дискретним компонентом $\{(S_i, \nu_i, \psi_i); i \geq 0\}$, де $\psi_i = (\psi_{1,i}, \psi_{2,i}, \psi_{3,i})$ – вектор довжин черг по потоках у момент τ_i. Для процесу Λ, ґрунтуючись на рівностях (3.3) – (3.5), має місце таке рекурентне співвідношення:

$$(\tau_{i+1}, S_{i+1}, \nu_{i+1}, \psi_{i+1}) = \begin{pmatrix} \tau_i + U(S_i), u(S_i, \psi_{1,i}, \eta_{1,i}), \\ \phi(\nu_i, \omega_i), \max\{\psi_i + \eta_i - \zeta_i, \overline{0}\} \end{pmatrix}, \qquad (3.8)$$

де $\eta_i = (\eta_{1,i}, \eta_{2,i}, \eta_{3,i}), \zeta_i = (\zeta_{1,i}, \zeta_{2,i}, \zeta_{3,i}), \overline{0} = (0,0,0)$.

Зазначимо, що векторне співвідношення $\psi_{i+1} = \max\{\psi_i + \eta_i - \zeta_i, \overline{0}\}$ передбачає виконання рівностей $\psi_{j,i+1} = \max\{\psi_{j,i} + \eta_{j,i} - \zeta_{j,i}, 0\}$ при $i \geq 0, \quad j = 1,2,3$. Беручи до уваги обрану нами екстремальну стратегію обслуговування α_O, одержимо такий вираз:

$$\psi_{j,i+1} = \psi_{j,i} + \eta_{j,i} - \overline{\zeta}_{j,i} = \psi_{j,i} + \eta_{j,i} - \min\{\psi_{j,i} + \eta_{j,i}, \zeta_{j,i}\} =$$
$$= \max\{\psi_{j,i} + \eta_{j,i} - \zeta_{j,i}, 0\}$$

Для вивчення ймовірнісних властивостей мітки $\{(S_i, \nu_i, \psi_i); i \geq 0\}$ зупинимося на деяких властивостях умовних розподілів величин $\eta_{j,i}$ і $\zeta_{j,i}$. Припустимо, що в цій моделі при фіксованих значеннях мітки $\{(S_k, \nu_k, \psi_k); k = \overline{0,i}\}$ випадкові величини $\eta_{j,i}$ й $\zeta_{j,i}$ незалежні та їх умовні розподіли при кожному $i \geq 0$ й при $y = 1,2,k$ задовольняють співвідношенням:

$$P(\eta_{j,i} = y \mid S_k = S^{(r_k)}, \nu_k = c^{(s_k)}, \psi_{j,k} = x_{j,k}; 0 \leq k \leq i) = P(\eta_{j,i} = y \mid S_i = S^{(r_i)}, \nu_i = c^{(s_i)})$$
$$P(\zeta_{j,i} = z \mid S_k = S^{(r_k)}, \nu_k = c^{(s_k)}, \psi_{j,k} = x_{j,k}; 0 \leq k \leq i) = P(\zeta_{j,i} = z \mid S_i = S^{(r_i)}, \nu_i = c^{(s_i)})$$

$y, z = 0,1,k; c_k = 0,1; r = \overline{1,7}; x_{j,k} = \{0,1,\dots k\}$.

Відповідно можна одержати аналітичні вирази для стану випадкового середовища $c^{(0)}$ й $c^{(1)}$.

Для $c^{(0)}$:

$$P\left(\eta_{j,i} = y \,\middle|\, S_i = S^{(r)}, \nu_i = c^{(0)}\right) = \frac{(\sigma_j \cdot T_r)^y}{(y!) \cdot e^{\sigma_j \cdot T_r}}, \qquad (3.9)$$

де $\sigma_j > 0$ – інтенсивність надходження заявок по потоку k_j.

Для $c^{(1)}$ з деякими спрощеннями одержимо:

$$\mathrm{P}\left(\eta_{j,i} = y \,\middle|\, S_i = S^{(r)}, \nu_i = c^{(1)}\right) = \frac{[A1 + A2]}{e^{(\sigma'_j \cdot T_r)}}, \qquad (3.10)$$

де $A1 = \dfrac{(\sigma'_j \cdot T_r)^y \cdot (1 - B_{g_j})^y}{(y!)}$, $A2 = \sum_{\xi=0}^{y-2} \dfrac{(\sigma'_j \cdot T_r)^\xi \cdot (1 - B_{g_j})^\xi}{(\xi!)}$;

$$\sigma'_j = \frac{\sigma_j}{1 + \dfrac{B_{g_j}}{1 - B_{q_j}}};$$

B_{g_j}, B_{q_j} - параметри розподілу Бартлетта [11].

Відповідні залежності, отримані в ході моделювання станів $c^{(0)}$ й $c^{(1)}$, показані на рис. 3.4.

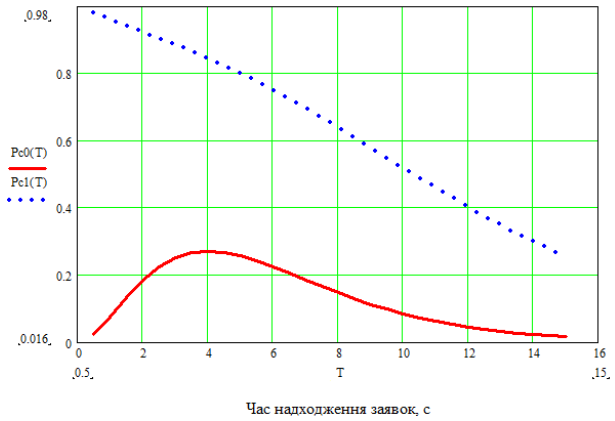


Рис. 3.4. Результати моделювання стану $c^{(0)}$ й $c^{(1)}$

Відповідно до запропонованого у другому розділі роботи методу інтелектуального розпізнавання загроз, складемо базу знань для процедури складання вирішального правила при КНІ DoS/DDoS при неоднорідних потоках запитів у ІКС, див. табл. 3.1.

Таблиця 3.1

Ознаки при КНІ DoS/DDoS у ІКС

Частковий параметр стану ІС	Універсум	Терми для лінгвістичної оцінки
ϕ_1 – інтенсивність потоку кадрів (запитів), що поступають к серверам ІС	[10 – 6000], кадр/с	немає (н), незначна кількість (нк), середня кількість (ск), велика кількість (вк)
ϕ_2 – номінальна пропускна спроможність середовища передачі даних ІС	[10,100], Мбіт/с	низка (нпс), середня (спс), велика (впс)
ϕ_3 – кількість спроб доступу до середовища передачі даних із урахуванням генерованих нападником	$[0, N_a]$	зафіксовані СЗІ (зф), незафіксовані СЗІ(нф)
ϕ_4 – час очікування обслуговування транзакції	[0,001 – 0,01], с	неприйнятний (нп), середній (ср), нормальний (н)
ϕ_5 – довжина пакету	[1 – 65529], байт	низька (н), нижче за критичну (нкр), критична (кр), вище за критичну (вкр)
ϕ_6 – кількість великих пакетів при атаці типу Ping of Death	[0,1], у. о.	мала кількість (м), середня (с), велика (в)

ϕ_7 – кількість http-запитів на об'єкт атаки	[0,1], у. о.	мала кількість (м), середня (с), велика (в)
ϕ_8 – наявність TCP flood пакетів	[0,1], у. о.	мала кількість (м), середня (с), велика (в)
ϕ_9 – наявність UDP flood пакетів	[0,1], у. о.	мала кількість (м), середня (с), велика (в)
ϕ_{10} – наявність ICMP пакетів	[0,1], у. о.	мала кількість (м), середня (с), велика (в)
ϕ_{11} – міжкадровий інтервал	[10 – 100], біт	малий (м), середній (с), великий (в)
ϕ_{12} – наявність HTTP POST пакетів	[0,1], у. о.	мала кількість (м), середня (с), велика (в)
ϕ_{13} – наявність HTTP GET пакетів	[0,1], у. о.	мала кількість (м), середня (с), велика (в)
ϕ_{14} – наявність HTTP flood пакетів	[0,1], у. о.	мала кількість (м), середня (с), велика (в)
…		
ϕ_z - інші фактори		

Наприклад, скорочено систему логічних рівнянь, що відповідає співвідношенню для інтелектуального розпізнавання низькоактивних КНІ DoS/DDoS прикладного рівня («повільний» HTTP GET flood і «повільний» HTTP POST flood), запишемо так:

$$\mu^{d_j}(\mathrm{S}) = \bigvee_{p=1}^{h_j}\left[\mu^{y_1^{jp}}(y_1) \wedge \mu^{\phi_я^{jp}}(\phi_{13}) \wedge \mu^{\phi_я^{jp}}(\phi_{14})\right],\ p = \overline{1, h_j}\ , j = \overline{1, M}\ ,$$

де $\mu^{y_1^{jp}}(y_1)$, $\mu^{\phi_я^{jp}}(\phi_{13})$, $\mu^{\phi_я^{jp}}(\phi_{14})$ – функції належності змінних y_1, ϕ_{13}, ϕ_{14} до їх нечітких термів y_1^{jp}, ϕ_{13}^{jp}, ϕ_{14}^{jp} відповідно;

S – стан захисту ІКС від атак DoS/DDoS;

y_1 – стан ІБ {нижче за критичний (нкр), критичний (кр), вище за критичний (вкр), високий (в)};

$\vee$ – логічне АБО, $\wedge$ – логічне І, як операції *max* і *min* відповідно.

Згідно зі структурою аналізованої ІКС або АСК з вхідними неоднорідними (конфліктними) потоками вимог (даних), найбільший інтерес становить дослідження процесів обслуговування по потоках k_1 і k_3.

Ключову властивість дискретного компонента процесу Λ можна сформулювати у вигляді наступної теореми [16].

Теорема: Послідовності $\{(S_i, \nu_i, \psi_i), i \geq 0\}$, $\{(S_i, \nu_i, \psi_{1,i}), i \geq 0\}$ і $\{(S_i, \nu_i, \psi_{1,i}, \psi_{3,i}), i \geq 0\}$ при заданому розподілі вектора (S_0, ν_0, ψ_0) є Марковськими.

Згідно з визначенням [16], дана послідовність буде Марковською, якщо виконана рівність:

$$P\left(S_{i+1} = S^{(r)}, \nu_{i+1} = c^{(s)}, \psi_{i+1} = x_{i+1} \middle| \mathrm{A}\right) = P\left(S_{i+1} = S^{(r)}, \nu_{i+1} = c^{(s)}, \psi_{i+1} = x_{i+1} \middle| \mathrm{B}\right), \tag{3.11}$$

де $\mathrm{A} = \{\omega : S_k = S^{(r_k)}, \nu_k = c^{(s_k)}, \psi_k = x_k; 0 \leq k \leq i\}$,

$\mathrm{B} = \{\omega : S_i = S^{(r_i)}, \nu_i = c^{(s_i)}, \psi_i = x_i\}$.

Застосовуючи формулу повної ймовірності і прийняті в моделі основні властивості її випадкових елементів, одержимо:

$$\mathrm{P}\left(S_{i+1} = S^{(r)}, \nu_{i+1} = c^{(s)}, \psi_{i+1} = x_{i+1} \middle| A\right) = \sum_{y_1, y_2, y_3 = 0}^{\infty} \sum_{z_1, z_2, z_3 = 0}^{\infty} \mathrm{P}\left(\begin{array}{l} S_{i+1} = S^{(r)}, \nu_{i+1} = c^{(s)}, \quad \psi_{i+1} = x_{i+1}, \eta_i \\ = (y_1, y_2, y_3), \ \zeta_i = (z_1, z_2, z_3) \middle| A \end{array}\right) \tag{3.12}$$

Після спрощення одержимо такий вираз, що описує ймовірнісні стани системи:

$$\mathrm{P}\left(S_{i+1}=S^{(r)},\nu_{i+1}=c^{(s)},\psi_{i+1}=x_{i+1}\,|A \right)$$

$$=\sum_{y_1,y_2,y_3=0}^{\infty}\sum_{z_1,z_2,z_3=0}^{\infty}\left[\begin{array}{l}\mathrm{P}\left(\eta_i=(y_1,y_2,y_3)\,\middle|\,S_{i+1}=S^{(r_i)},\nu_{i+1}=c^{(s_i)}\right)\cdot\\ \cdot\mathrm{P}\left(\zeta_i=(z_1,z_2,z_3)\,\middle|\,S_{i+1}=S^{(r_i)},\nu_{i+1}=c^{(s_i)}\right)\cdot\\ \cdot\mathrm{P}\left(u\left(S^{(r_i)},x_{1,i},y_1\right)=S^{(r)}\right)\cdot\mathrm{P}\left(\phi_i\left(c^{(s_i)},\omega_i\right)=c^{(s)}\right)\cdot\\ \cdot\mathrm{P}\left(max\left\{\psi_{j,i}+\eta_{j,i}-\zeta_{j,i},0\right\}=x_{j,i+1};j=\overline{1,3}\,\middle|\begin{array}{l}A,\eta_i=(y_1,y_2,y_3),\\ \zeta_i=(z_1,z_2,z_3),\\ S_{i+1}=S^{(r)},\\ \nu_{i+1}=c^{(s)}\end{array}\right)\end{array}\right]$$

для правої частини рівності – такий вираз:

$$\mathrm{P}\left(S_{i+1}=S^{(r)},\nu_{i+1}=c^{(s)},\psi_{i+1}=x_{i+1}\,|B \right)=$$

$$=\sum_{y_1,y_2,y_3=0}^{\infty}\sum_{z_1,z_2,z_3=0}^{\infty}\left[\begin{array}{l}\mathrm{P}\left(\eta_i=(y_1,y_2,y_3)\,\middle|\,S_{i+1}=S^{(r_i)},\nu_{i+1}=c^{(s_i)}\right)\cdot\\ \cdot\mathrm{P}\left(\zeta_i=(z_1,z_2,z_3)\,\middle|\,S_{i+1}=S^{(r_i)},\nu_{i+1}=c^{(s_i)}\right)\cdot\\ \cdot\mathrm{P}\left(u\left(S^{(r_i)},x_{1,i},y_1\right)=S^{(r)}\right)\cdot\mathrm{P}\left(\phi_i\left(c^{(s_i)},\omega_i\right)=c^{(s)}\right)\cdot\\ \cdot\mathrm{P}\left(max\left\{x_{j,i}+y_j-z_j,0\right\}=x_{j,i+1};\,j=\overline{1,3}\right)\end{array}\right] \qquad (3.13)$$

Таким чином, випадкова послідовність подій у системі $\left\{\left(S_i,\nu_i,\psi_{1,i}\right);i\geq 0\right\}$ утворює ланцюг Маркова з нескінченним числом станів.

Дослідимо властивості одномірних розподілів:

$$= \begin{pmatrix} P\left(S_i = S^{(r_i)}, \nu_i = c^{(s_i)}, \psi_i = x_i\right): \\ S^{(r_i)} \in S, c^{(s_i)} \in \left(c^{(0)}, c^{(1)}\right), x_i = \left(x_{1,i}, x_{2,i}, x_{3,i}\right), x_{j,i} \in (0,1,\ldots), j = 1,2,3 \end{pmatrix}, i \geq 0. \quad (3.14)$$

Тут початковий розподіл P_0 вважається заданим. Одержимо рекурентні співвідношення виду $P_{i+1} = P_i \cdot P$, де P – нескінченно вимірна матриця перехідних ймовірностей за один крок процесу $\{(S_i, \nu_i, \psi_{1,i}); i \geq 0\}$.

Докладно розглянемо ймовірнісні властивості послідовностей $\{(S_i, \nu_i, \psi_{1,i}); i \geq 0\}$ і $\{(S_i, \nu_i, \psi_{1,i}, \psi_{3,i}); i \geq 0\}$.

Ґрунтуючись на раніше отриманих залежностях, неважко одержати такі, рекурентні за $i \geq 0$, співвідношення для цих послідовностей:

$$\left(S_{i+1}, \nu_{i+1}, \psi_{1,i+1}\right) = \begin{pmatrix} u\left(S_i, \psi_{1,i}, \eta_{1,i}\right), \Theta_i\left(\nu_i, \omega_i\right), \\ \max\left\{\psi_{1,i} + \eta_{1,i} - \zeta_{1,i}, 0\right\} \end{pmatrix},$$

$$\left(S_{i+1}, \nu_{i+1}, \psi_{1,i+1}, \psi_{3,i}\right) = \begin{pmatrix} u\left(S_i, \psi_{1,i}, \eta_{1,i}\right), \Theta_i\left(\nu_i, \omega_i\right), \\ \max\left\{\psi_{1,i} + \eta_{1,i} - \zeta_{1,i}, 0\right\}, \\ \max\left\{\psi_{3,i} + \eta_{3,i} - \zeta_{3,i}, 0\right\} \end{pmatrix}.$$

Зазначимо, що дослідження послідовностей $\left((S_i, \nu_i, \psi_i); i \geq 0\right)$ і $\left((S_i, \nu_i, \psi_{1,i}, \psi_{2,i}); i \geq 0\right)$ відбувається аналогічно.

Уведемо такі позначення:

$$P\left(S_i = S^{(r_i)}, \nu_i = c^{(s_i)}, \psi_{1,i} = x\right) = Q_i\left(S^{(r)}, c^{(s)}, x\right),$$

$$P\left(S_i = S^{(r_i)}, \nu_i = c^{(s_i)}, \psi_{1,i} = x, \psi_{3,i} = y\right) = Q_i\left(S^{(r)}, c^{(s)}, x, y\right),$$

$$P\left(\eta_{j,i} = z \middle| S_i = S^{(r)}, \nu_i = c^{(s)}\right) = \phi_{j,s}\left(z, T_r\right), \qquad (3.15)$$
$$j = 1,2,3,\ r = \overline{1,7},\ s = 0,1.$$

На підставі доведеної властивості марковості розглянутих послідовностей і формули повної ймовірності одержимо таку залежність:

$$Q_{i+1}\left(S^{(r)}, c^{(s)}, w_1, w_3\right) = \sum \left[\begin{array}{l} Q_i\left(S^{(k)}, c^{(h)}, x, y\right) \cdot \\ \cdot P\left(\begin{array}{l} S_{i+1} = S^{(r)}, \nu_{i+1} = c^{(s)}, \\ \psi_{1,i+1} = w_1, \psi_{3,i+1} = w_3 \end{array} \middle| \begin{array}{l} S_i = S^{(k)}, \nu_i = c^{(h)}, \\ \psi_{1,i} = x, \psi_{3,i} = y \end{array} \right) \end{array} \right], \qquad (3.16)$$

де підсумовування ведеться за $\left(S^{(k)}, c^{(h)}, x, y\right) \in S \cdot \left\{c^{(0)}, c^{(1)}\right\} \cdot \{0,1,\ldots\} \cdot \{0,1,\ldots\}$.

Тепер обчислимо умовні ймовірності:

$$\mathrm{P}\left(S_{i+1} = S^{(r)}, \nu_{i+1} = c^{(s)}, \psi_{1,i+1} = w_1, \psi_{3,i+1} = w_3 \middle| S_i = S^{(k)}, \nu_i = c^{(h)}, \psi_{1,i} = x, \psi_{3,i} = y\right) =$$
$$= \sum_{n_1,n_3}^{\infty} \sum_{j_1,j_3}^{\infty} \left[\mathrm{P}\left(\begin{array}{l} \eta_{1,i} = n_1, \eta_{3,i} = n_3, \zeta_{1,i} = j_1, \zeta_{3,i} = j_3, \\ S_{i+1} = S^{(r)}, \nu_{i+1} = c^{(s)}, \psi_{1,i+1} = w_1, \psi_{3,i+1} = w_3 \end{array} \middle| \begin{array}{l} S_i = S^{(k)}, \nu_i = c^{(h)}, \\ \psi_{1,i} = x, \psi_{3,i} = y \end{array} \right) \right] =$$

$$= \sum_{n_1,n_3}^{\infty} \sum_{j_1,j_3}^{\infty} \left[\begin{array}{l} \mathrm{P}\left(\eta_{1,i} = n_1 \middle| S_i = S^{(k)}, \nu_i = c^{(h)}\right) \cdot \mathrm{P}\left(\eta_{3,i} = n_3 \middle| S_i = S^{(k)}, \nu_i = c^{(h)}\right) \cdot \\ \cdot \mathrm{P}\left(\zeta_{1,i} = j_1 \middle| S_i = S^{(k)}, \nu_i = c^{(h)}\right) \cdot \mathrm{P}\left(\zeta_{3,i} = j_3 \middle| S_i = S^{(k)}, \nu_i = c^{(h)}\right) \cdot \\ \cdot \mathrm{P}\left(u\left(S^{(k)}, x, n_1\right) = S^{(r)}\right) \cdot \\ \cdot \mathrm{P}\left(\Theta_i\left(c^{(h)}, \varpi_i\right) = c^{(s)}\right) \cdot \mathrm{P}\left(max\{x + n_1 - j_1, 0\} = w_1\right) \cdot \\ \cdot \mathrm{P}\left(max\{y + n_3 - j_3, 0\} = w_3\right) \end{array} \right]. \quad (3.17)$$

Тоді остаточно одержимо таку залежність:

$$Q_{i+1}\left(S^{(r)}, c^{(s)}, w_1, w_3\right)=$$

$$= \sum \left[\begin{array}{l} Q_i\left(S^{(k)}, c^{(h)}, x, y\right)\cdot \phi_{1,h}\left(n_1, \mathrm{T}_k\right)\cdot \phi_{3,h}\left(n_3, \mathrm{T}_k\right)\cdot \\ \cdot \mathrm{P}\left(\zeta_{1,i} = j_1 \middle| S_i = S^{(k)}, \nu_i = c^{(h)}\right)\cdot \\ \cdot \mathrm{P}\left(\zeta_{3,i} = j_3 \middle| S_i = S^{(k)}, \nu_i = c^{(h)}\right)\cdot \mathrm{P}\left(u\left(S^{(k)}, x, n_1\right) = S^{(r)}\right)\cdot \\ \cdot \mathrm{P}\left(\Theta_i\left(c^{(h)}, \varpi_i\right) = c^{(s)}\right)\cdot \\ \cdot \mathrm{P}\left(\max\left\{x + n_1 - j_i, 0\right\} = w_1\right)\cdot \\ \cdot \mathrm{P}\left(\max\left\{y + n_3 - j_3, 0\right\} = w_3\right) \end{array} \right]. \qquad (3.18)$$

Тут підсумовування ведеться за всіма точками:

$$\left(S^{(k)}, c^{(h)}, x, y, n_1, n_2, j_1, j_3\right) \in S\cdot\left\{c^{(0)}, c^{(1)}\right\}\cdot\{0,1,\ldots\}\cdot\{0,1,\ldots\}\cdot\{0,1,\ldots\}\cdot\{0,1,\ldots\}\cdot$$
$$\cdot\{0,1,\ldots\}\cdot\{0,1,\ldots\}.$$

Враховуючи вид умовних розподілів для $\eta_{1,i}, \eta_{3,i}, \zeta_{1,i}, \zeta_{3,i}$, можна одержати конкретний вид рекурентних формул для одномірних розподілів дискретного компонента $\left\{\left(S_i, \nu_i, \psi_{1,i}, \psi_{3,i}\right); i \geq 0\right\}$.

Нижче докладно наведений тільки висновок формули для ймовірностей $Q_{i+1}\left(S^{(r)}, c^{(s)}, w_1, w_3\right)$ при $r \neq 2, r \neq 6, r \neq 7$.

Використовуючи останній вираз і враховуючи, що при $r \neq 2, r \neq 6, r \neq 7$ на інтервалах часу $[\tau_i, \tau_{i+1})$ жоден із потоків заявок з АРМ, МАРМ у ІКС не обслуговується, одержимо для $w_1, w_3 \in \{0,1,\ldots\}, s \in \{0,1\}$ таку рекурентну залежність:

$$Q_{i+1}\left(S^{(r)}, c^{(s)}, w_1, w_3\right)=$$

$$= \mathrm{P}_{0,s}\sum_{x=0}^{\infty}\sum_{y=0}^{\infty}\left[Q_i\left(S^{(r-1)}, c^{(0)}, x, y\right)\cdot \phi_{1,0}\left(w_1 - x, \mathrm{T}_{r-1}\right)\cdot \phi_{3,0}\left(w_3 - y, \mathrm{T}_{r-1}\right)\right] +$$
$$+ \mathrm{P}_{1,s}\sum_{x=0}^{\infty}\sum_{y=0}^{\infty}\left[Q_i\left(S^{(r-1)}, c^{(1)}, x, y\right)\cdot \phi_{1,1}\left(w_1 - x, \mathrm{T}_{r-1}\right)\cdot \phi_{3,1}\left(w_3 - y, \mathrm{T}_{r-1}\right)\right] \qquad (3.19)$$

де вважаємо при $r = 1 : S^{(r-1)} = S^{(6)}, T_{r-1} = T_6$.

Ймовірності $P_{h,s}, (h,s) \in \{0,1\} \cdot \{0,1\}$ утворюють матрицю: $L = \begin{bmatrix} 1-a & a \\ b & 1-b \end{bmatrix}$.

Далі через $l_{1,s}, l_{3,s}, l'_{3,s}$ будемо позначати відповідно цілі частини величин $\mu_{1,s}\mathrm{T}_1, \mu_{3,s}\mathrm{T}_5, \mu_{3,s}\mathrm{T}_7$, де $\mu_{j,s}$ – інтенсивність обслуговування по потоку k_j, якщо система перебуває в стані $c^{(s)}$.

Оскільки при звичайному вході штатного користувача в систему $S_i = S^{(1)}$ обслуговуються вимоги потоку k_1, рекурентні співвідношення для ймовірностей $Q_{i+1}(S^{(2)}, c^{(s)}, w_1, w_3)$ при $w_1 \geq 0, w_3 \geq 0, s \in \{0,1\}$ отримуємо у вигляді:

$$Q_{i+1}(S^{(2)}, c^{(s)}, 0, w_3) = \sum_{h=0}^{1} P_{h,s} \sum_{x=0}^{l_{1,h}} \sum_{y=0}^{w_3} Q_i(S^{(1)}, c^{(h)}, x, y) \cdot \\ \cdot \phi_{3,h}(w_3 - y, T_1) \cdot \sum_{n_1=0}^{l_{1,h}-x} \phi_{1,h}(n_1, T_1), \tag{3.20}$$

$$Q_{i+1}(S^{(2)}, c^{(s)}, w_1, w_3) = \sum_{h=0}^{1} P_{h,s} \sum_{x=0}^{w_1+l_{1,h}} \sum_{y=0}^{w_3} Q_i(S^{(1)}, c^{(h)}, x, y) \cdot \\ \phi_{1,h}(w_1 + l_{1,h} - x, T_1) \cdot \phi_{3,h}(w_3 - y, T_1), \\ \textit{для} \quad w_1 \geq 1. \tag{3.21}$$

При КНІ у ІКС зловмисник створює ситуацію, за якої $S_i \in \{S^{(5)}, S^{(7)}\}$ відбувається обслуговування вимог по потоку k_3, то при $r = 6$ одержимо, що $Q_{i+1}(S^{(6)}, c^{(s)}, 0, w_3) = 0$ при всіх $w_3 \geq 0$ та $i \geq 0$, а при $w_1 \geq 1$ одержимо таку залежність:

$$Q_{i+1}(S^{(6)}, c^{(s)}, w_1, 0) = \\ = \sum_{h=0}^{1} \mathrm{P}_{h,s} \left[\begin{array}{l} \sum_{x=0}^{w_1} \sum_{y=0}^{l_{3,h}} Q_i(S^{(5)}, c^{(h)}, x, y) \cdot \phi_{1,h}(w_1 - x, \mathrm{T}_5) \cdot \sum_{n_3=0}^{l_{3,h}-y} \phi_{3,h}(n_3, \mathrm{T}_5) + \\ + \sum_{x=0}^{w_1} \sum_{y=0}^{l'_{3,h}} Q_i(S^{(7)}, c^{(h)}, x, y) \cdot \phi_{1,h}(w_1 - x, \mathrm{T}_7) \cdot \sum_{n_3=0}^{l'_{3,h}-y} \phi_{3,h}(n_3, \mathrm{T}_7) \end{array} \right], \tag{3.22}$$

а при будь-яких $w_3 \geq 1$:

$$Q_{i+1}\left(S^{(6)},c^{(s)},w_1,w_3\right)=$$
$$=\sum_{h=0}^{1}\mathrm{P}_{h,s}\left[\begin{array}{l}\sum_{x=0}^{w_1}\sum_{y=0}^{w_3+l_{3,h}}Q_i\left(S^{(5)},c^{(h)},x,y\right)\cdot\phi_{1,h}\left(w_1-x,\mathrm{T}_5\right)\cdot\\ \cdot\phi_{1,h}\left(w_3+l_{3,h}-y,\mathrm{T}_5\right)+\\ +\sum_{x=0}^{w_1}\sum_{y=0}^{w_3+l'_3}Q_i\left(S^{(7)},c^{(h)},x,y\right)\cdot\phi_{1,h}\left(w_1-x,\mathrm{T}_7\right)\cdot\\ \cdot\phi_{1,h}\left(w_3+l'_{3,h}-y,\mathrm{T}_7\right)\end{array}\right]. \quad (3.23)$$

Для ймовірностей $Q_{i+1}\left(S^{(7)},c^{(s)},w_1,w_3\right)$ одержимо $Q_{i+1}\left(S^{(7)},c^{(s)},w_1,w_3\right)=0$ при будь-якому $w_1\geq 0, i\geq 0,\ s\in\{1,0\}$:

$$Q_{i+1}\left(S^{(7)},c^{(s)},0,0\right)=$$
$$=\sum_{h=0}^{1}P_{h,s}\left[\begin{array}{l}\sum_{y=0}^{l_{3,h}}Q_i\left(S^{(5)},c^{(h)},0,y\right)\cdot\phi_{1,h}\left(0,T_5\right)\cdot\sum_{n_3=0}^{l_{3,h}-y}\phi_{3,h}\left(n_3,T_5\right)+\\ +\sum_{y=0}^{l'_{3,h}}Q_i\left(S^{(7)},c^{(h)},0,y\right)\cdot\phi_{1,h}\left(0,T_7\right)\cdot\sum_{n_3=0}^{l'_{3,h}-y}\phi_{3,h}\left(n_3,T_7\right)\end{array}\right], \quad (3.24)$$

а при будь-яких $w_3\geq 0,\ s\in\{1,0\}$:

$$Q_{i+1}\left(S^{(7)},c^{(s)},0,w_3\right)=$$
$$=\sum_{h=0}^{1}P_{h,s}\left[\begin{array}{l}\sum_{y=0}^{w_3+l_{3,h}}Q_i\left(S^{(5)},c^{(h)},0,y\right)\cdot\phi_{1,h}\left(0,T_5\right)\cdot\phi_{3,h}\left(w_3+l_{3,h}-y,T_5\right)+\\ +\sum_{y=0}^{w_3+l'_{3,h}}Q_i\left(S^{(7)},c^{(h)},0,y\right)\cdot\phi_{1,h}\left(0,T_7\right)\cdot\phi_{3,h}\left(w_3+l'_{3,h}-y,T_7\right)\end{array}\right]. \quad (3.25)$$

Варто зазначити, що оскільки ймовірності $Q_i\left(S^{(6)},c^{(s)},0,w_3\right)=0$ для $s\in\{1,0\}, w_3\geq 0,\ i\geq 0$, то з виразу (3.18) безпосередньо випливає, що $Q_i\left(S^{(1)},c^{(s)},0,w_3\right)=0$ при всіх для $s\in\{1,0\}, w_3\geq 0,\ i>0$.

Особливий інтерес становить ситуація, коли зловмисникові вдалося подолати рубіж МЕ й одержати доступ до ресурсів ІКС. Тобто, відповідно до графа, зображеного на рис. 3.3, подальші дії атакуючого випливають зі стану

системи S^5. Для моделювання КНІ у ІКС та АСК розглянуті додаткові потоки, які надсилаються зараженим абонентам на інші термінали, що перебувають у цей момент часу на зв'язку з вихідним терміналом.

Якщо припустити, що в початковий момент часу $\tau = 0$ число цілей PA для нападу на інформацію не є випадковою величиною й уразливості YZ_i, що ведуть до цілей, фіксовані в системі, то одержати доступ до уразливостей можна тільки після успішного проведення нападу на інформацію на уразливості, розташовані на нижніх рівнях графа атаки. Цей процес можна представити як процес розмноження уразливостей і надалі використовувати в нашій моделі для опису станів системи. Оскільки в реальних умовах КНІ вплив зловмисника відбуваються лише у фіксовані моменти часу, то уразливість YZ_i, існуючи до моменту часу $\tau - 0$, у момент часу τ породжує π_{YZ}^k нащадків. У разі навмисного КНІ у ІКС та АСК руйнувальні впливи майже завжди досягають мети. Однак наявність ресурсів (на цьому ми зупинимося в наступному розділі) дозволяє відновлювати систему. Тут також доцільно розглянути два варіанти:

1) вплив порушує працездатність усієї системи, при цьому, залежно від збитку, потрібен різний середній час на її відновлення;

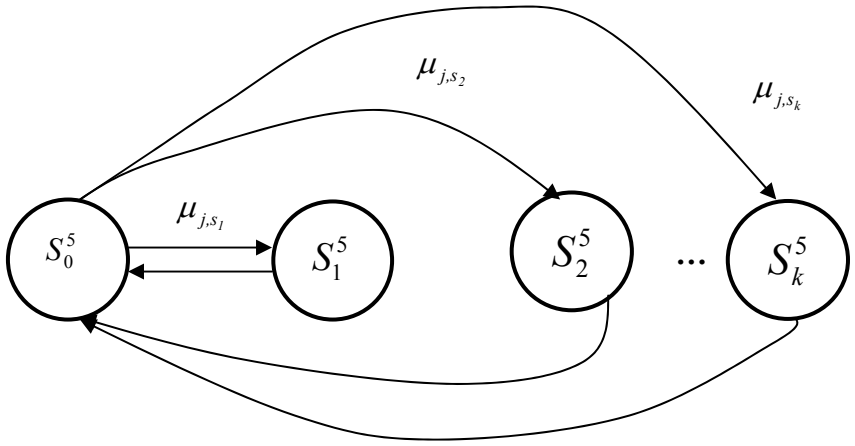
2) виводяться з ладу один або більше компонентів системи, система атакується, доки існують працездатні компоненти.

Звідси приходимо до оцінки інтенсивності обслуговування запитів та знаходимо оптимальні значення інтенсивностей відновлення у разі навмисного руйнуючого впливу на ІКС. Оцінка найкращої досяжної живучості наведена на рис. 3.5.

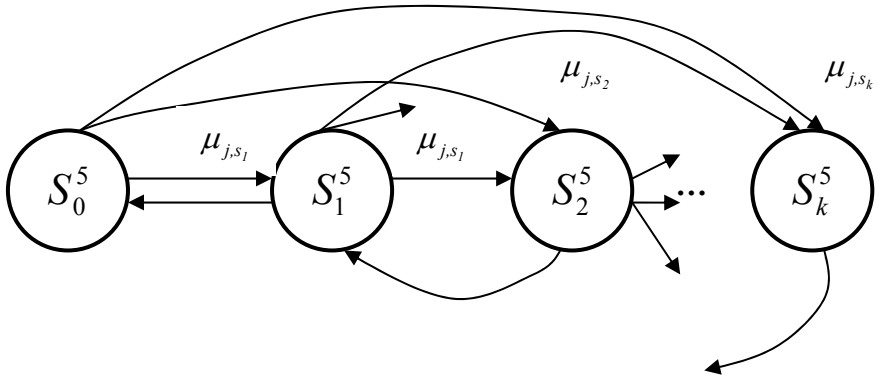
У моделі (варіант 1) застосуємо неординарний стаціонарний стохастичний процес, описуваний відповідної діаграмою станів (див. рис. 3.5 а).

Для варіанта 2 наведена діаграма станів на рис. 3.5 б. Оскільки мережа працездатна, якщо вижила хоча б одна її компонента, то в станах $1, ..., S^k - 1$ ІКС та АСК працездатна і продовжує зазнавати атакувального впливу. У стані S^k

система непрацездатна. Отже, розв'язуючи систему рівнянь, що описують ці стани системи, можна знайти ймовірність працездатності ІКС як $1-Q_k$.



а)



б)

а) варіант 1 - вплив зловмисника порушує працездатність ІКС, при цьому, залежно від збитку, потрібен різний середній час на її відновлення; б) зловмисник виводить з ладу один або більше компонентів ІКС, доки існують працездатні компоненти.

Рис. 3.5. Графи станів системи

Відповідно одержимо такі залежності ймовірнісних станів ІКС або АСК з урахуванням ДПРЗ ІБ і правила інтелектуального розпізнавання загроз у випадку нападу на інформацію $gov(p_{ax_i})$:

$$Q'_{i+1}\left(S^{(r)},c^{(s)},w_1,w_3,gov(p_{ax_i})\right)=$$

$$=\sum_{h=0}^{1}\sum_{x=0}^{\infty}\sum_{y=0}^{\infty}\sum_{n_1,n_3=0}^{\infty}\sum_{j_1,j_3=0}^{\infty}\left[\begin{array}{l} Q_i\left(S^{(r-1)},c^{(h)},x,y,gov(p_{ax_i})\right)\cdot\phi_{1,h}\left(n_1,\mathrm{T}_{r-1}\right)\cdot\phi_{3,h}\left(n_3,\mathrm{T}_{r-1}\right)\cdot \\ \cdot\mathrm{P}\left(\zeta_{1,i}=j_1\middle|S_i=S^{(r-1)},\nu_i=c^{(h)}\right)\cdot \\ \cdot\mathrm{P}\left(\zeta_{3,i}=j_3\middle|S_i=S^{(r-1)},\nu_i=c^{(h)}\right)\cdot \\ \cdot\mathrm{P}\left(\Theta_i\left(c^{(h)},\varpi_i\right)=c^{(s)}\right)\cdot\mathrm{P}\left(max\{x+n_1-j_i,0\}=w_1\right)\cdot \\ \cdot\mathrm{P}\left(max\{y+n_3-j_3,0\}=w_3\right) \end{array}\right] \quad (3.26)$$

та

$$\mathrm{P}\left(S_{i+1}=S^{(r)},\nu_{i+1}=c^{(s)},\psi_{i+1}=x_{i+1}\middle|B,gov(p_{ax_i})\right)=$$

$$=\sum_{y_1,y_2,y_3=0}^{\infty}\sum_{z_1,z_2,z_3=0}^{\infty}\left[\begin{array}{l} \mathrm{P}\left(\eta_i=(y_1,y_2,y_3)\middle|B\right)\cdot\mathrm{P}\left(\zeta_i=(z_1,z_2,z_3)\middle|B\right)\cdot \\ \cdot\mathrm{P}\left(S_{i+1}=S^{(r)}\middle|B,\eta_i=(y_1,y_2,y_3),\ \zeta_i=(z_1,z_2,z_3)\right)\cdot \\ \cdot\mathrm{P}\left(\nu_{i+1}=c^{(s)}\middle|B,\eta_i=(y_1,y_2,y_3),\zeta_i=(z_1,z_2,z_3),S_{i+1}=S^{(r)}\right)\cdot \\ \cdot\mathrm{P}\left(\psi_{i+1}=x_{i+1}\middle|\begin{array}{l}B,\eta_i=(y_1,y_2,y_3),\zeta_i=(z_1,z_2,z_3)\\ ,S_{i+1}=S^{(r)},\ \nu_{i+1}=c^{(s)}\end{array}\right) \end{array}\right] \quad (3.27)$$

Уточнимо тепер структуру ланцюга Маркова $\left\{\left(S_i,\nu_i,\psi_{1,i},\psi_{3,i}\right);i\geq 0\right\}$. Позначимо через $X=\left\{\left(S^{(r)},c^{(s)},w_1,w_3\right):r=1,2,\ldots,7;\ s=0,1;\ w_1,w_3=0,1,\ldots\right\}$.

Сформулюємо та доведемо два допоміжні твердження, що стосуються загальної структури ланцюга й асимптотичної поведінки розподілу розглянутого ланцюга Маркова при $i\to\infty$.

Лема. Простір X станів ланцюга Маркова $\left\{\left(S_i,\nu_i,\psi_{1,i},\psi_{3,i}\right);i\geq 0\right\}$ розпадається на незамкнену множину X_0 несуттєвих станів і мінімально замкнену множину X_1 істотних сполучених неперіодичних станів.

Доказ. З того, що $l_{1,s} > 0, l_{3,s} > 0, l'_{3,s} > 0, 0 < a, b < 1$ й $\phi_{1,s}(0, \mathrm{T}_r), \phi_{3,s}(0, \mathrm{T}_r) > 0$ для всіх $r = \overline{1,7}, s = 0,1$, випливає, що випадковий процес $\{(S_i, \nu_i, \psi_{1,i}, \psi_{3,i}); i \geq 0\}$ за деяке кінцеве число кроків з довільного стану $(S^{(r)}, c^{(s)}, w_1, w_3) \in X$ з позитивною ймовірністю по ланцюгу $(S^{(r)}, c^{(s)}, w_1, w_3) \to \ldots$ $\to (S^{(5)}, c^{(s)}, 0, w_3) \to (S^{(7)}, c^{(0)}, 0, 0)$ потрапить у стан $(S^{(7)}, c^{(0)}, 0, 0)$. Отже, стан $(S^{(7)}, c^{(0)}, 0, 0)$ є суттєвим.

Згідно з теоремою з [13], сукупність станів ланцюга, що сполучаються з $(S^{(7)}, c^{(1)}, 0, 0)$, також є суттєвою. Використовуючи отримані нами рекурентні співвідношення (3.13) – (3.19) і наведені вище зауваження неважко описати множину X'_0 так:

$$X'_0 = \{(S^{(1)}, c^{(s)}, 0, w_3): w_3 \geq 0, s = 0,1\} \cup \{(S^{(6)}, c^{(s)}, 0, w_3): w_3 \geq 0, s = 0,1\} \cup$$
$$\cup \{(S^{(7)}, c^{(s)}, w_1, w_3): w_1 \geq 0, w_3 \geq 0, s = 0,1\} \subset X_0.$$

Покажемо, що X_0 не містить інших станів, крім відмічених. Візьмемо, наприлад, стан $(S^{(1)}, c^{(0)}, 1, w_3)$, де $w_3 \geq 0$. По ланцюжку переходів $(S^{(7)}, c^{(0)}, 0, 0) \to$ $\to (S^{(6)}, c^{(0)}, 1, w_3) \to (S^{(1)}, c^{(0)}, 1, w_3)$ ланцюг Маркова $\{(S_i, \nu_i, \psi_{1,i}, \psi_{3,i}); i \geq 0\}$ перейде з суттєвого стану $(S^{(7)}, c^{(0)}, 0, 0)$ в стан $(S^{(1)}, c^{(0)}, 1, w_3)$. Отже, стан $(S^{(1)}, c^{(0)}, 1, w_3)$ є суттєвим і сполучуваним з $(S^{(7)}, c^{(0)}, 0, 0)$. Зазначений перехід мо-жливий з позитивною ймовірністю, оскільки $(1-a) \cdot \phi_{1,0}(1, \mathrm{T}_7) \cdot \phi_{3,0}(w_3 + l'_{3,s}, \mathrm{T}_7) > 0$ та $(1-a) \cdot \phi_{1,0}(1, \mathrm{T}_6) \cdot \phi_{3,0}(0_{3,s}, \mathrm{T}_6) > 0$. Аналогічно доводиться, що можливим є перехід з $(S^{(7)}, c^{(0)}, 0, 0)$ або $(S^{(7)}, c^{(1)}, 0, 0)$ у будь-який інший стан, що не належать множині X'_0. Отже, $X'_0 = X_0$. Оскільки стан $(S^{(7)}, c^{(0)}, 0, 0)$ при досить високому рівні кваліфікації зловмисника є досяжним з будь-якого стану $(S^{(r)}, c^{(s)}, w_1, w_3) \in X$, то множина X_0 не є замкненою, а множина X містить єдине замкнене мінімальне $X_1 = X \setminus X_0$. З очевидної нерівності

$$P\left(S_{i+1}=S^{(7)},\nu_{i+1}=c^{(0)},\psi_{1,i+1}=0,\psi_{3,i+1}=0\middle|S_i=S^{(7)},\nu_i=c^{(0)},\psi_{1,i}=0,\psi_{3,i}=0\right)\geq \quad (3.28)$$
$$\geq (1-a)\cdot\phi_{1,0}(0,T_7)\cdot\phi_{3,0}(0,T_7)>0$$

випливає, що всі стани з X будуть неперіодичними.

При будь-якому початковому розподілі $\left\{Q_0\left(S^{(r)},c^{(s)},w_1,w_3\right):\left(S^{(r)},c^{(s)},w_1,w_3\right)\in X\right\}$ векторного ланцюга Маркова $\left\{\left(S_i,\nu_i,\psi_{1,i},\psi_{3,i}\right),i\geq 0\right\}$ або для всіх $\left(S^{(r)},c^{(s)},w_1,w_3\right)\in X$:

$\lim\limits_{i\to\infty}Q_i\left(S^{(r)},c^{(s)},w_1,w_3\right)=0$ і в системі не існує стаціонарного розподілу або існують межі:

$\lim\limits_{i\to\infty}Q_i\left(S^{(r)},c^{(s)},w_1,w_3\right)=Q\left(S^{(r)},c^{(s)},w_1,w_3\right)\geq 0$ такі, що $\sum Q\left(S^{(r)},c^{(s)},w_1,w_3\right)=1$, і в системі існує стаціонарний розподіл.

Зі структури множини X_0 та з того, що $\min\left\{\cdot\phi_{1,0}(1,T_6),\phi_{3,0}(0,T_6),\cdot\phi_{1,1}(1,T_6)\cdot\phi_{3,1}(0,T_6)\right\}=\phi>0$, випливає, що випадковий процес $\left\{\left(S_i,\nu_i,\psi_{1,i},\psi_{3,i}\right),i\geq 0\right\}$ із довільного стану $\left(S^{(r)},c^{(s)},w_1,w_3\right)\in X_0$ з позитивною ймовірністю не меншою ніж ϕ за один крок може досягти множини X_1. Позначимо через $P_0\left(S^{(r)},c^{(s)},w_1,w_3\right)$ ймовірність того, що розглянутий ланцюг Маркова, виходячи з довільного несуттєвого стану $\left(S^{(r)},c^{(s)},w_1,w_3\right)\in X_0$ коли-небудь досягне деякого суттєвого стану з X_1. Відомо, що величини $P_0\left(S^{(r)},c^{(s)},w_1,w_3\right)\left(S^{(r)},c^{(s)},w_1,w_3\right)\in X_0$ є рішеннями системи рівнянь, наведеної в [66, 182, 195]. Тоді, у силу нерівності $\phi>0$ й леми 1, ця система є цілком регулярною й має обмежене розв'язання $P_0\left(S^{(r)},c^{(s)},w_1,w_3\right)=1$, $\left(S^{(r)},c^{(s)},w_1,w_3\right)\in X_0$. У цьому можна переконатися безпосередньою підстановкою.

Отже, асимптотична поведінка одномірного розподілу $\left\{Q_i\left(S^{(r)},c^{(s)},w_1,w_3\right):\left(S^{(r)},c^{(s)},w_1,w_3\right)\in X\right\}$ випадкового векторного процесу

$\{(S_i, \nu_i, \psi_{1,i}, \psi_{3,i}), i \geq 0\}$ при $i \to \infty$ не залежить від початкового розподілу $\{Q_0(S^{(r)}, c^{(s)}, w_1, w_3) : (S^{(r)}, c^{(s)}, w_1, w_3) \in X\}$.

Таким чином, процес реалізації загрози становить послідовність переміщень, реалізованих у вигляді півкроків за ланцюгом Маркова. При цьому ланцюг перебуває в кожному стані деякий випадковий час, що визначений відповідною до цього стану щільністю розподілу ймовірності для випадкового часу перебування, і потім виконується крок і перевірка логічних умов перемикання ланцюга в наступний стан. Послідовність станів ланцюга (графа атаки) і буде траєкторією модельованого процесу.

3.2. Моделі оцінки ймовірності реалізації загроз інформаційно-комунікаційному середовищу

Особливістю застосування мереж Петрі-Маркова (МПМ) і ланцюгів Маркова для оцінки реалізації загрози є той факт, що поряд із використанням напівмарковської матриці $Q(\tau)$, яка характеризує часові та стохастичні параметри моделі, і вектора P, що описує ймовірність появи станів процесу в початковий момент часу, вводиться матриця логічних умов LY, елементи якої рівні [11]:

$$\nu_{j(z),i(a)} = \begin{cases} L_p[S_{1(a),j(z)},...,S_{i(a),j(z)},...S_{k(a),j(z)}], \textit{якщо } a_{i(a)} \in O_A(z_{j(z)}); \\ 0, \qquad\qquad \textit{якщо } a_{i(a)} \notin O_A(z_{j(z)}). \end{cases}$$

У цьому випадку функція L_p – це логічна функція, що дозволяє виконання півкроків з переходів у стани згідно зі структурою мережі Петрі, де півкроку від позиції з номером $i_{(a)}$ до переходу з номером $j_{(z)}$,

$$S_{i(a),j(z)} = (a_{i(a)}, z_{j(z)}), 1(a) \leq i(a) \leq K(a),$$

визначає умови спрацьовування переходу $z_{j(z)}$ в позицію $a_{j(a)} \in O_z(a_{j(a)})$. Два послідовні півкроки утворюють крок. Крім того, особливість такої мережі, що відрізняє її від звичайної мережі Петрі, полягає в тому, що кожний перехід спрацьовує тільки з певною ймовірністю.

Напівмарковська матриця представляє собою добуток матриці ймовірностей переходів $\{o_{ij}\}$ і матриці щільності ймовірностей часів перебування процесу в кожному *i -м* стані ІКС $\{f_{ij}(\tau)\}$, якщо вважати, що сам перехід відбувається миттєво, тобто $Q(\tau) = \{o_{ij} \cdot f_{ij}(\tau)\}$.

Таким чином, процес реалізації загрози – це послідовність переміщень, реалізованих у вигляді півкроків по МПМ, при цьому МПМ перебуває в кожному стані деякий випадковий час, що визначено відповідною до цього стану щільністю розподілу ймовірності для випадкового часу перебування, і потім виконується півкрок і перевірка логічних умов перемикання мережі в наступний стан. Аналітичний опис процесу здійснюється, як і для звичайних напівмарковських процесів, у вигляді інтегро-диференціальних рівнянь за траєкторіями переміщень із початкового стану в кінцевий [15].

Нехай $h(tr : 1(a) \to j(a) = h(tr_1)$ – номер траєкторії переміщення зі стану $a_{1(a)}$ (тут індекс із буквою означає номер стану) у стан $a_{j(a)}$, що містить послідовність півкроків зі стану в перехід, потім з переходу у стан і т.д.:

$$S_{1[h(tr)]},\ S_{2[h(tr)]},\ldots,\ S_{i[h(tr)]},\ldots,\ S_{j[h(tr)]},$$

де *i, j* – індекси, які відповідають номеру стану або номеру переходу, що входять до обраної траєкторії $h(tr)$. Кількість таких траєкторій дорівнює $H(tr)$. Ймовірність і щільність розподілу часу виконання відповідного півкроку визначаються як $P_{j(a)j(z)}$ і $f_{j(a)j(z)}$. Тоді ймовірність і щільність

розподілу часу переміщення зі стану $a_{1(a)}$ в стан $a_{j(a)}$ за траєкторією $h(tr_{1j})$ визначається з таких співвідношень [15]:

$$P_{h(tr_{1j})} = \prod_{j[h(tr_{1j})]=1}^{J[h(tr_{1j})]} P_{j[h(tr_{1j})]};$$

$$f_{h(tr_{1j})} = f_{1[h(tr_{1j})]} * f_{2[h(tr_{1j})]} * \ldots * f_{i[h(tr_{1j})]} * \ldots * f_{J[h(tr_{1j})]},$$

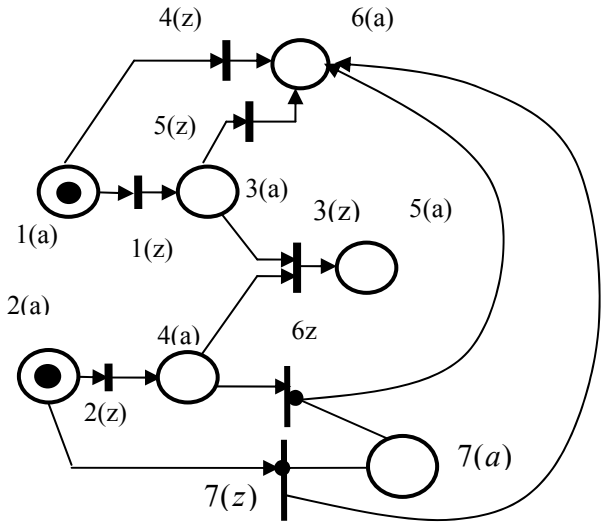
де $J[h(tr_{1j})]$ – загальна кількість позицій і переходів у траєкторії $h(tr_{1j})$;

$*$ – операція згортки стану $a_{1(a)}$ за всіма можливими траєкторіями зі співвідношень:

$$P_{1(a)j(a)} = \prod_{h(tr_{1j})=1}^{H(tr_{1j})} P_{h(tr_{1j})}; \tag{3.29}$$

$$f_{1(a)j(a)} = \frac{\prod_{h(tr_{1j})=1}^{H(tr_{1j})} P_{h(tr_{1j})} \cdot f_{h(tr_{1j})}}{\prod_{h(tr_{1j})=1}^{H(tr_{1j})} P_{h(tr_{1j})}}. \tag{3.30}$$

Розглянемо як приклад моделі КНІ «Аналіз мережного трафіка» і КНІ DoS/DDoS, представлених у вигляді мережі Петрі та графа із використанням ланцюгів Маркова (див. рис. 3.6 – 3.10).

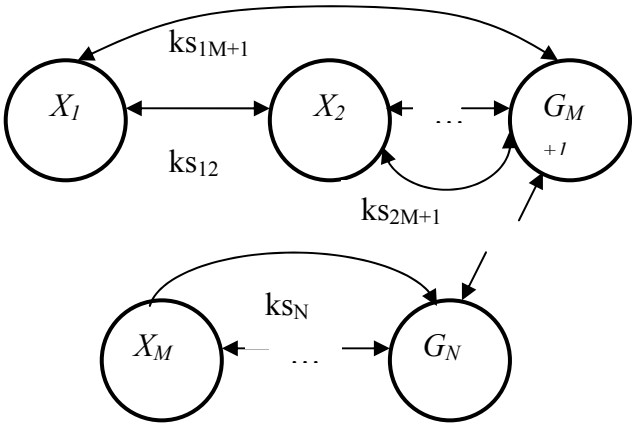


1(a) – атакований хост увімкнувся в мережу загального користування (МЗК); 2(a) – хост зловмисника увімкнувся в МЗК; 3(a) – атакований хост установив з'єднання з АРМ, МАРМ і почав обмін даними; 4(a) – зловмисник одержав доступ до сервера; 5(a) – зловмисник одержав ім'я користувача і пароль; 6(a) – КНІ зірвано; 7(a) – умови для перехоплення пакетів відсутні; 1(z) – установлення з'єднання між хостом і абонентом; 2(z) – одержання зловмисником доступу до сервера; 3(z) – аналіз трафіка; 4(z), 5(z), 6(z) і 7(z) – зрив КНІ

Рис. 3.6. Мережа Петрі, що описує сигнатуру атаки типу «Аналіз мережного трафіка» (при відсутності захисту)

Тут півкрок $S_{3(z)} \to S_{5(a)}$ може бути виконаний тільки у разі виконання півкроків $S_{3(a)} \to S_{3(z)}$.

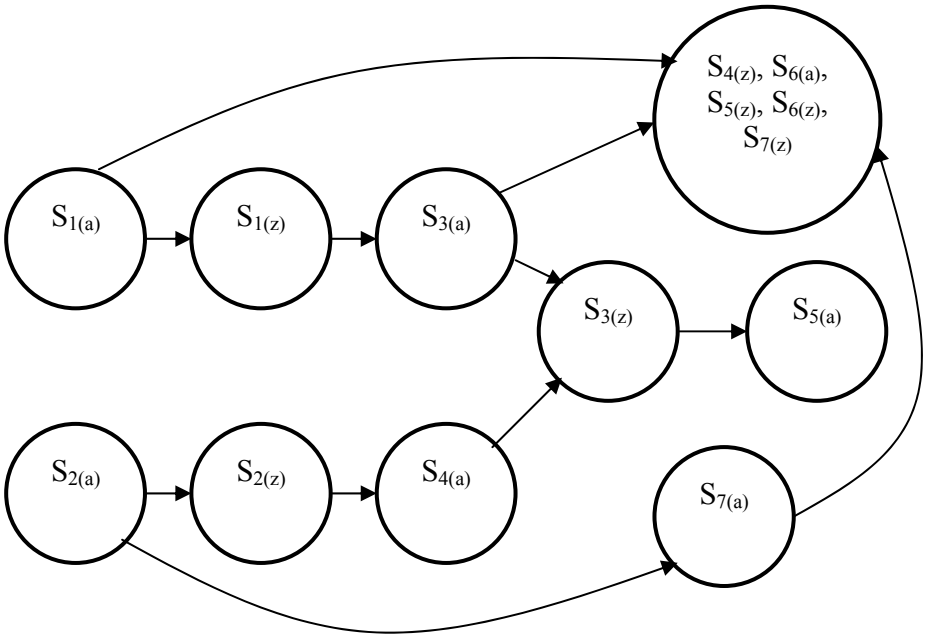
Оскільки перехід системи з одного стану в інший (наприклад, $S_{3(z)} \to S_{5(a)}$) спрацьовує миттєво, то динаміка спрацьовування ланцюга Маркова або мережі Петрі-Маркова (МПМ) визначається тільки ймовірностями спрацьовування ланцюга (переміщення зі стану в стан) і щільностями розподілу часу перебування процесу в кожному стані. Тоді, у даному прикладі досить розглянути процес системи за двома траєкторіями зі станів $S_{1(a)}$ і $S_{2(a)}$ у перехід $S_{3(z)}$.



$X = \{x_i \mid i = 1..M\}$ – множина хостів; $G = \{g_i \mid j = M+1..N\}$ – множина маршрутизаторів (роутерів); $KS = \{ks_{KL} \mid k = 1..N, L = 1..N\}$ – множина ліній зв'язку на мережному рівні

ks_{12} – атакований хост увімкнувся в мережу загального користування (МЗК); ks_{2M+1} - атакований хост установив з'єднання з абонентом і почав обмін даними (наявність трафіка хоста з абонентом); ks_{NM} – зловмисник одержав доступ до сервера (хоста), через який проходить трафік атакованого хоста; і т.д.

а)



$S_{1(a)}$ –, атакований хост увімкнувся в МЗК; $S_{2(a)}$ – хост зловмисника ввімкнувся в МЗК; $S_{3(a)}$ – атакований хост установив з'єднання з абонентом і почав обмін даними; $S_{4(a)}$ – зловмисник одержав доступ до сервера; $S_{5(a)}$ – зловмисник одержав ім'я користувача й пароль; $S_{7(a)}$ – умови для перехоплення пакетів відсутні; $S_{1(z)}$ – установлення з'єднання між хостом і абонентом; $S_{2(z)}$ – одержання зловмисником доступу до сервера, через який проходить трафік атакованого хоста; $S_{3(z)}$ – аналіз трафіка (перехоплення пакетів атакованого хоста); $S_{6(a)}$, $S_{4(z)}$, $S_{5(z)}$, $S_{6(z)}$, $S_{7(z)}$ – зрив атаки

б)

Рис. 3.7. Графи атаки типу «Аналіз мережного трафіка», змодельовані ланцюгом Маркова

Якщо в графові є переходи з одного стану в інший з логічними умовами, то тоді ланцюг розбивається на ділянки, що відповідають:

а) переміщенням з початкового стану до першого такого переходу;

б) станам між переходами;

в) станам після переходів до кінцевого стану (рис. 3.8).

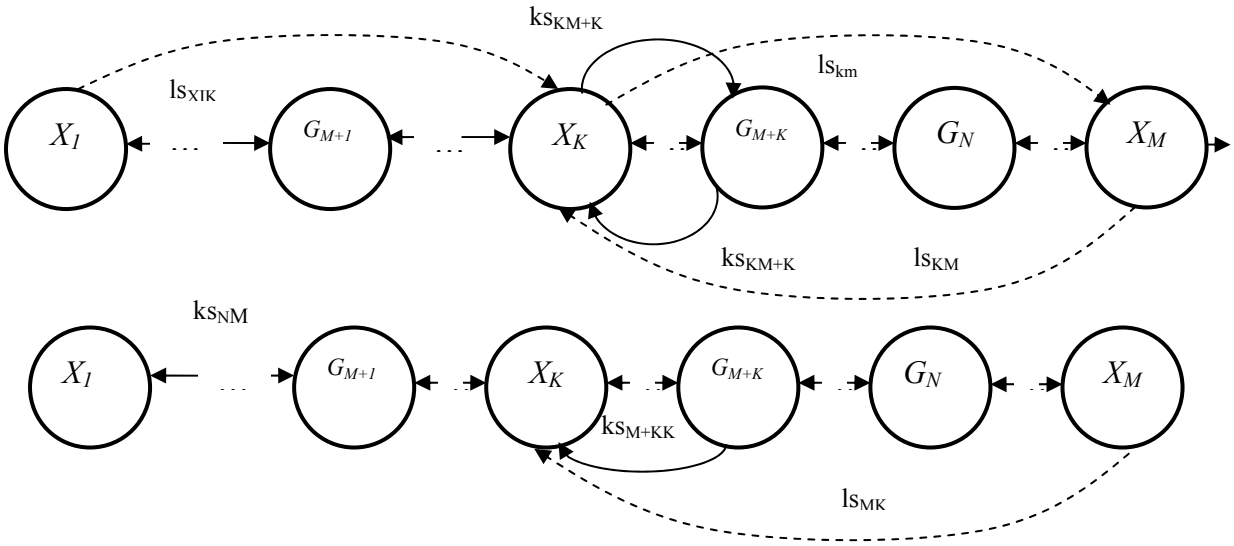
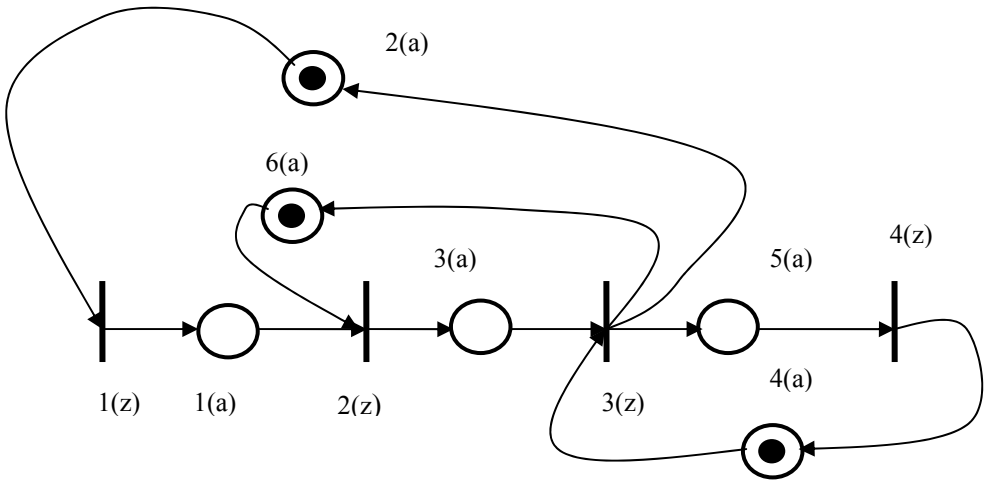


Рис. 3.8. Граф КНІ DoS/DDoS у ІКС



1(a) – число заглушених каналів, що очікують пошуку; 2(a) – число активних каналів; 3(a) – число каналів, що очікують перенаправлення; 4(a) – число вільних запасних каналів; 5(a) – число заглушених неактивних каналів; 6(a) – кількість можливостей пошуку заглушених каналів; 1(z) – атака виявлена; 2(z) – пошук заглушеного каналу завершений; 3(z) – перенаправлення пакетів на вільний канал завершено; 4(z) – канал відновлений

Рис. 3.9. Мережа Петрі КНІ DoS/DDoS у ІКС

Ймовірність переміщення процесу з початкового стану $S_{i(a)}$ в кінцевий $S_{j(z)}$ за траєкторією $h(tr_{ij})$ визначається на підставі розв'язання системи інтегро-диференціальних рівнянь звичайного вигляду [7]:

$$\Phi_{ij}(h(tr_{ij}),\tau) = o_{ij} \cdot \int_0^{\tau} f_{ik}(h(tr_{ij}),\tau) \cdot \Phi_{kj}(h(tr_{ij}) - \tau) \cdot d\tau. \quad (3.31)$$

При цьому вважають, що оскільки траєкторія обрана (дуги графа відомі), то альтернативні варіанти переміщення по дугах, інцидентні позиції $S_{i(a)}$, не розглядаються.

Однак, якщо на траєкторії $h(tr_{ij})$ існує перехід з логічною умовою та на цьому переході існують кілька траєкторій, то необхідно розрахувати ймовірність того, що цей логічний перехід спрацює.

Нехай загальний номер (за нумерацію МПМ або ланцюга Маркова) такого переходу позначений як ϑ, а поточний номер цього переходу відповідно до нумерації траєкторії $h(tr_{ij})$ відповідає величині $n_{h(rt_{ij})}$, тоді зазначена ймовірність визначається зі співвідношення:

$$\Phi_{\vartheta}(\tau) = \begin{cases} \prod\limits_{h(tr)\in H} \Phi_{i+n_{h(tr_{ij})}}(h(tr_{ij}),\tau) - \text{для} \quad \wedge; \\ 1 - \prod\limits_{h(tr)\in H} \left[1 - \Phi_{i+n_{h(tr_{ij})}}(h(tr_{ij}),\tau)\right] - \text{для} \quad \vee, \end{cases} \quad (3.32)$$

де перехідні ймовірності $\Phi_{i+n_{h(tr_{ij})}}(h(tr_{ij}),\tau)$ визначаються на підставі розв'язання системи рівнянь, що описують систему до переходу з логічною умовою.

Якщо більше логічних переходів за траєкторією $h(tr_{ij})$ не існує, то ймовірність того, що процес до розрахункового моменту часу досягне кінцевого переходу, а отже, потрапить в останню вершину графа, знаходимо так:

$$\Phi_{i,j(t)} = \prod_{k=1}^{j} o_{spm(S_a)}, k(s_a, j) \cdot \int_0^t \Phi_{s_a}(\tau) \cdot f_{spm(s_a),j}(t-\tau) \cdot d\tau, \quad (3.33)$$

де $spm(s_a)$ – номер позиції по МПМ, що йде безпосередньо за переходом під номером s_a;

$k(s_a, j)$ – номер переходу по порядку при переміщенні по МПМ від переходу з номером s_a до переходу з номером j.

Ймовірність $\Phi_{i,j(t)}$ є, по суті, ймовірністю реалізації загрози. Якщо за даною траєкторією є ще переходи з логічними умовами, то для них викладена процедура повторюється. Розрахунки за зазначеними формулами є досить громіздким, тому на практиці доцільно застосовувати пуассоновське наближення для щільності розподілу ймовірностей часу переміщення в переходи МПМ. Для звичайних переходів такий час розраховується шляхом диференціювання в точці 0 відповідної характеристичної функції. Розрахунки показують, що помилка в оцінці ймовірності реалізації загрози при заміні довільного розподілу часу переміщення на пуассоновське призводить до помилок, які не перевищують десяти відсотків.

Розглянемо, як можна розрахувати середній час переміщення в перехід з логічними умовами. Нехай є дві траєкторії, які сходяться на переході з логічною умовою «І» ($\wedge$), і щільності розподілу часу переміщення до даного переходу за обома траєкторіями розподілені приблизно за експонентними законами з параметрами $\bar{t}_1$ й $\bar{t}_2$. Тоді щільність розподілу ймовірності для часу від початку процесу до спрацьовування переходу з логічною умовою «І» та «АБО» ($\vee$) визначається зі співвідношень:

$$f_{1\wedge 2}(\tau) = f_1(\tau)\cdot RF_2(\tau) + f_2(\tau)\cdot RF_1(\tau);$$
$$f_{1\vee 2}(\tau) = f_1(\tau)\cdot[1 - RF_2(\tau)] + f_2(\tau)\cdot[1 - RF_1(\tau),$$

де $RF_1(\tau), RF_2(\tau)$ – функції розподілу часу від початку процесу до спрацьовування переходу з логічною умовою.

При цьому математичне очікування зазначеного часу для переходу з логічною умовою «І» розраховується так:

$$M\hbar_{(\bar{t}_{1\wedge 2})} = \int_0^\infty \tau\cdot[\, f_1(\tau)\cdot RF_2(\tau) + f_2(\tau)\cdot RF_1(\tau)]\cdot d\tau,$$

аналогічно для переходу з логічною умовою «АБО»:

$$M\hbar_{(\bar{t}_{1\vee 2})} = \int_0^\infty \tau\cdot\{\, f_1(\tau)\cdot[1 - RF_2(\tau)] + f_2(\tau)\cdot[1 - RF_1(\tau)]\}\cdot d\tau.$$

У разі експонентного наближення формули мають такий вигляд:

$$M\hbar_{(\bar{t}_{1\wedge 2})} = \frac{\bar{t}_1^2 + \bar{t}_1\cdot\bar{t}_2 + \bar{t}_2^2}{\bar{t}_1 + \bar{t}_2} \quad \text{та} \quad M\hbar_{(\bar{t}_{1\vee 2})} = \frac{\bar{t}_1\cdot\bar{t}_2}{\bar{t}_1 + \bar{t}_2}, \qquad (3.34)$$

де $\bar{t}_1, \bar{t}_2$ – середній час переміщення в перехід з логічною умовою за першою і другою траєкторією відповідно.

Великою перевагою МПМ є можливість аналізу таких властивостей паралельних процесів, як безпека, активність, схоронність, досяжність. Завдання моделювання КНІ може бути сформульовано таким чином: дана мережа Петрі *PN*, що моделює атаковану ІКС або АСК; потрібно доповнити вихідну МПМ елементами, що моделюють процес нападу на інформацію, і визначити досяжність у знов отриманій мережі Петрі стану, що відповідає

досягненню мети нападу на інформацію, або активність переходів вихідної мережі з урахуванням впливу внесених елементів.

Проведений аналіз дозволив визначити, що для вирішення завдань дослідження часових параметрів модельованих процесів доцільно використовувати розширення формалізму МПМ, відому як *E*-Мережі. Формально модель *E* задається у вигляді:

$$E = \{A_e, Z_e, I_e, O_e, G_e\},$$

де A_e – множина позицій, Z_e – множина переходів, I_e – множина вхідних функцій переходів, O_e – множина вихідних функцій переходів, G_e – множина глобальних змінних моделі.

Приклад *E*-Мережної моделі «Атака – хибний об'єкт ІКС» представлена на рис. 3.10.

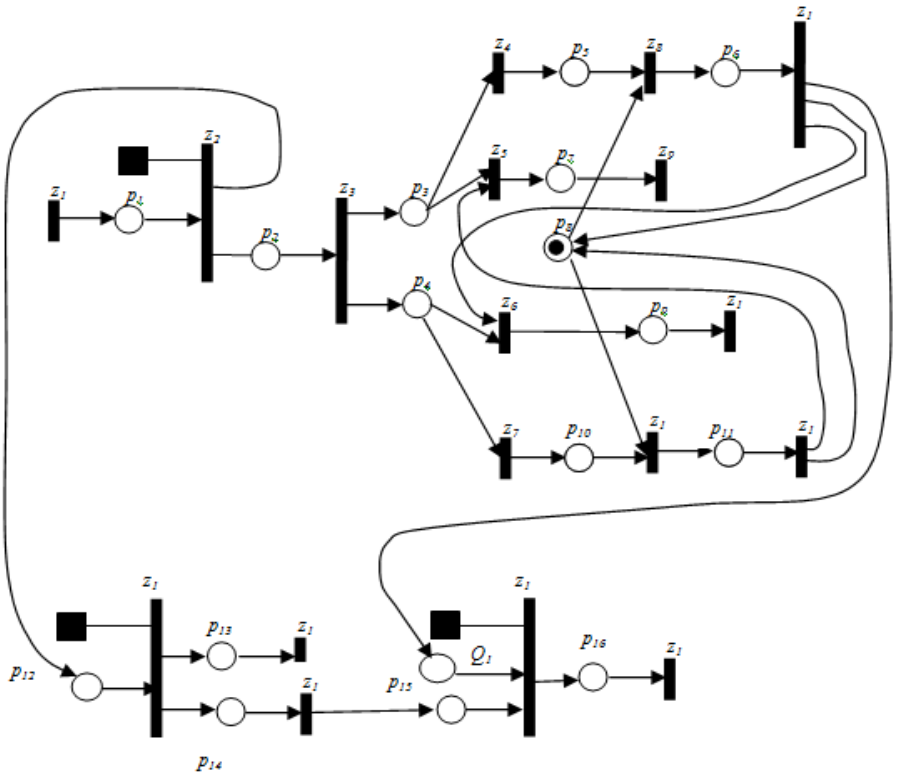


Рис. 3.10. Модель КНІ «Хибний об,єкт ІКС»

Припустимо, що ми маємо справу з різними видами нападу на інформацію з метою одержання несанкціонованого доступу до ІКС або АСК.

СЗІ покликана парирувати ці напади на інформацію, тим самим забезпечуючи захист від НСД. Розглянемо два види її відмов: приховані відмови (тобто такі відмови системи безпеки, які не можуть бути виявлені без проведення профілактичних заходів) і хибні відмови (тобто хибне спрацьовування СЗІ). Таким чином, несанкціонований доступ до інформації із використанням певного класу КНІ, можливий тільки тоді, коли напад на інформацію припадає на період непрацездатності відповідної підсистеми СЗІ, тобто – «аварії» (див. табл. 2.2).

Вважаємо, що існує NT_{TA} типів КНІ, і при цьому атаки різних видів статистично незалежні. Припустимо, що атаки на ІКС і їх парирування можна описувати за допомогою процесу відновлення ПЗ та ІМ. Для середнього часу до аварії СЗІ ІКС та АСК можна записати вираз вигляду:

$$TM_w = TM_{\tau'} + (\frac{1 - P_{r2}}{P_{r2}}) \cdot TM_\tau , \qquad (3.35)$$

де $TM_{\tau'}$ – середня тривалість циклу регенерації процесу функціонування системи, за умови, що сталася аварія внаслідок КНІ, с;

TM_τ – середня тривалість циклу регенерації процесу функціонування системи, за умови, що аварії внаслідок КНІ, с;

P_{r2} – ймовірність того, що на циклі регенерації розглянутого процесу відбулася аварія.

Використовуючи методи, аналогічні викладеним у [11], вдалося одержати досить компактні вирази для обчислення асимптотичних значень зазначених параметрів. Для врахування структури підсистеми системи безпеки використовується такий підхід. Передбачається, що кожний з елементів відмовляє самостійно, але відновлення проводиться тільки для підсистеми в цілому. Це дозволяє використовувати добре відомий у теорії надійності метод

шляхів і перетинів, і, в той же час, дискретно розглядати моменти регенерації процесу функціонування СЗІ.

При такому припущенні була отримана наступна експонентна оцінка для функції розподілу часу до першої аварії СЗІ внаслідок КНІ:

$$F_W(\tau) = 1 - e^{\frac{-P_{r2}\cdot\tau}{TM_\tau}}, \qquad (3.36)$$

за умови $\frac{TM^{\,2}_{\tau}\cdot P_{r2}}{(TM_r)^2} \to 0.$

Отже, запропонована модель дозволяє враховувати такі особливості функціонування системи захисту інформації, як хибне спрацьовування, періодичні перевірки, складну структуру підсистем системи безпеки.

Таким чином, описані моделі реалізації загроз ІКС або АСК не тільки становлять самостійний практичний інтерес, але і є прикладом можливої формалізації опису інших КНІ. Певна громіздкість проведених розрахунків ускладнює практичне застосування апарату мереж Петрі-Маркова й ланцюгів Маркова для моделювання розглянутих процесів. Однак в експонентному наближенні розрахунки ймовірності реалізації нападу на інформацію виявляються досить простим. Наведений підхід дозволяє перейти до кількісних процедур оцінки можливостей реалізації загроз у комп'ютерних мережах з урахуванням фактора часу й тим самим підвищити обґрунтованість проведених заходів із захисту інформації.

Враховуючи все вищесказане, наступним завданням дослідження було імітаційне моделювання різних режимів функціонування підсистем ІКС та АСК зі змінною структурою й неоднорідних потоками даних [10]. Цьому питанню присвячений розділ 4 навчального посібника.

Для тестування продуктивності ДПРЗ та розробленої експертної системи обрана задача зіставлення загрозам КНІ – DoS/DDoS, «Хибний об’єкт ІКС» та «Аналіз мережного трафіка» відомих методів захисту. База знань із 9 правил

розрізняє вище вказані атаки на основі відомих ознак, вхідних і додаткових атрибутів, що описують поточний стан системи (табл. 2.4, 2.5 та 3.1).

На рис. 3.11 – 3.13 показані основні результати, отримані в ході тестового моделювання процедур розпізнавання КНІ DoS/DDoS, «Хибний об’єкт ІКС» та «Аналіз мережного трафіка» на ІКС.

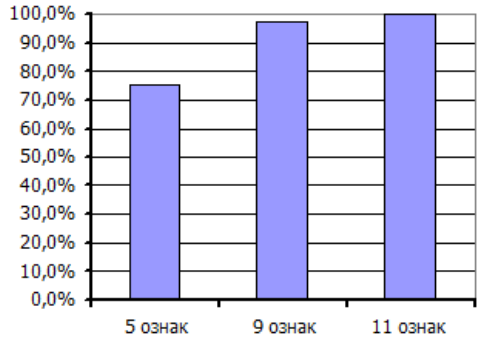


Рис. 3.11. Ймовірність виявлення КНІ DoS/DDoS

Рис. 3.12. Ймовірність виявлення КНІ «Хибний об’єкт ІКС»

Рис. 3.13. Ймовірність виявлення КНІ «Аналіз мережного трафіка»

Для проведення аналізу отриманих в розрахунків представимо отримані данні у вигляді таблиці, див. табл. 3.2.

Таблиця 3.2

Зведені результати моделювання станів системи у випадку КНІ

№	Тип атаки	Вихідні дані			
		Середній показник ймовірності розпізнавання аномального стану системи			
		Дерево рішень [2]	Граф атаки [3]	Алгоритм найближчого кластера [3]	ДПРЗ+ нечіткі бази знань
1	DoS/DDoS	0,75	0,89	0,86	0,94
2	Хибний об’єкт ІКС	0,63	0,72	0,66	0,85
3	Аналіз мережного трафіка	0,7	0,93	0,81	0,91

Проаналізувавши отримані дані, можна зробити висновок, що всі запропоновані в розділі моделі, надали досить точні результати. Ймовірність вирішення завдання розпізнавання загрози нападу на інформацію та створення небезпеки для ІКС склала 85–98%, залежно від типу атаки на ресурси ІКС або АСК.

З урахуванням раніше отриманих виразів 3.12, 3.13, 3.17, 3.21 і 3.22, розглянемо зміну ентропії системи при використанні в ній різних комбінацій довжини пакетів і різних варіантів обслуговування заявок при КНІ ІКС.

3.3. Визначення станів інформаційно-комунікаційного середовища на підставі мультимножин для інтелектуального розпізнавання загроз

Ентропію ІКС або АСК можна розглядати як кількість інформації, пов'язаної зі структурою системи і її станами [12]. Тобто, саме ентропія дозволяє судити про технічний стан ІКС, що працює у звичайних умовах і при впливі КНІ. Таким чином, ентропія може розглядатися як міра «структурованості» деякого стану S_i або міра віддаленості структури одного стану від іншого.

Тоді випадковий процес (ВП) у ІКС або АСК, що характеризує стан систем та функціонує в інтервалі часу від τ_0 до Т, описується вектором змінних стану ІБ:

$$S_i(\tau) = f(SX(\tau), he) + hl(\tau), \qquad (3.37)$$

де $he, hl(\tau)$ – «шуми» загальної природи;

$SX(\tau)$ – вектор змінних станів системи.

Спостереження величини $S_i(\tau)$ здійснюється в моменти часу $\tau_i = \tau_0 + j\Delta, \quad j = \overline{0,n,}$ із кроком дискретизації $\Delta > 0$ за імітаційною моделлю ІС або АКС в АПК.

Поставимо у відповідність кожному виділеному стану системи (для альтернативних гіпотез $A\Omega = \{A\Omega_1, ...A\Omega_N\}$, що становлять повну групу подій і фізично інтерпретують стану системи) мультимножину [10]:

$$M\Theta_i = \{l_{M\Theta_i}(s) \cdot s \mid s \in UK_{sig}, l_{M\Theta_i}(s) \in ZR\},$$

де $l_{M\Theta_i}(s)$ – функція числа екземплярів мультимножини, що визначає кратність елемента $s \in UK_{sig}$;

UK_{sig} – множина, потужність якого дорівнює максимальному рівню сигналу, характерного для ознаки об'єкта.

З урахуванням положень, наведених у розділі 2, узагальнимо основні етапи процедури розпізнавання:

1. Визначаємо характеристичні ознаки. Складемо для кожного вузла ІКС або АСК повну групу стану системи – $A\Omega = \{A\Omega_1, ...A\Omega_N\}$, яким будуть відповідати первинні специфікації $M\Theta_i$.

2. Визначимо оцінки розподілу ймовірностей P_{S_i} характерних для станів системи, у які вона потрапила внаслідок КНІ, і опишемо зміну ентропії усіх підсистем ESI_{S^*} за формулою:

$$ESI_{S^*} = -\sum_{i=1}^{max} P_{S_i} \cdot log_2 P_{S_i}. \qquad (3.38)$$

3. За результатами спостережень $(S^*_L = \{S^*(\tau), S^*(\tau+1), ..., S^*(\tau+L-1)\}$ формуємо випадкову стохастичну мультимножину:

$$M\Theta^*_L = \{uk^L_{sig1}, uk^L_{sig2}, ..., uk^L_{sigL}\}, \qquad (3.39)$$

де uk^L_{sigj} – загальне число зустрічальності сигналів, характерних для j-го стану системи;

L – контрольне «вікно».

4. Обчислюємо інформаційні відстані між мультимножинами $DIS(M\Theta_i, M\Theta^*_L)$ $(i = \overline{0,I})$ за $p_{ax} \geq 1$ ознаками відмінності.

5. Приймаємо рішення на користь стану, для якого величина $DIS(M\Theta_i, M\Theta_L^*)$ є найменшою для кожної ознаки p_{axi}. Одночасно обчислюємо вагові коефіцієнти окремих рішень:

$$kf_1^j = arg \min_{i=\overline{0,I}} DIS(M\Theta_i, M\Theta_L^*),$$
$$kf_2^j = arg \min_{i=0,I,\, i\neq i_1^j} DIS(M\Theta_i, M\Theta_L^*), (j=\overline{1,J}). \quad (3.40)$$

6. Обираємо відповідно до процедури голосування, описаній у розділі 2, той стан системи, для якого ваговий коефіцієнт більше:

$$kf_1 = arg \min_{i=\overline{0,I}} kf_1^j,$$
$$kf_2 = arg \min_{i=0,I,\, i\neq i_1^j} kf_2^j. \quad (3.41)$$

Розглянемо результати чисельного моделювання на прикладі розпізнавання станів реальних систем. Ентропія ІКС або АСК визначається співвідношенням:

$$ESI_{nv} = -\sum_{ib=min}^{max} P_{ib} \cdot log_2 P_{ib},$$

де nv – можливе число варіантів довжин пакета;

P_{ib} – ймовірність появи в каналі передачі даних пакета довжиною *ib* байт;

min, max – мінімальне та максимальне значення довжини пакета.

При відсутності нападу на інформацію ентропія системи при більших nv приблизно постійна величина, що рівна 4.

За допомогою генератора трафіка у пакеті MATLAB 7 та Simulink створювалися атаки 1 -*оїї* та 2 -*оїї* групи.

У першій групі моделювалися атаки, при яких використовуються мережні пакети однакової довжини.

У другій групі розглядалися атаки, при яких довжина пакета – випадкова величина в заданому інтервалі та, крім того, мережні пакети представляли собою потоки Бартлетта (потік пачок).

На рис. 3.14 наведена крива зміни ентропії залежно від довжини переданих мережних пакетів. Із графіка бачимо, що при збільшенні числа nv ентропія системи при впливі атакою 1 *-ої* групи швидко прагне до 0. Це виділяє DoS – атаку 1 *-ої* групи із процесів, що відбуваються в мережі.

На рис. 3.15 зображена ентропія для DoS – атаки 2 *-ої* групи. Ентропія також прагне до 0, але повільніше, ніж у першому випадку. Це пов'язано з тим, що ця DoS – атака здійснювалася потоком мережних пакетів випадкової довжини, рівномірно розподіленої на деякому інтервалі.

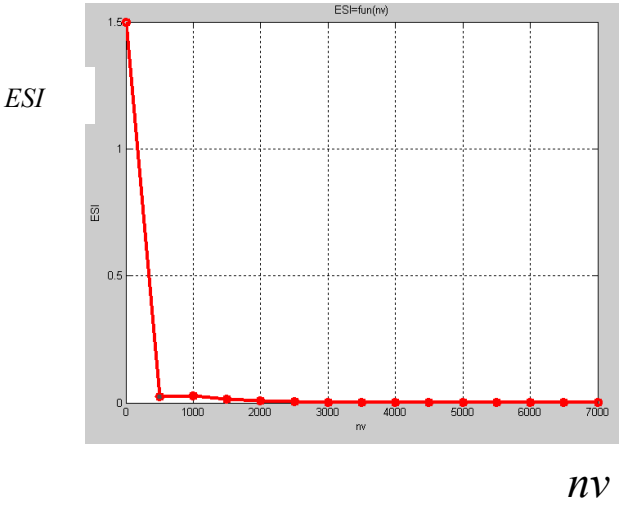


Рис. 3.14. Зміна ентропії системи залежно від довжини переданих мережних пакетів $ESI = f(nv)$

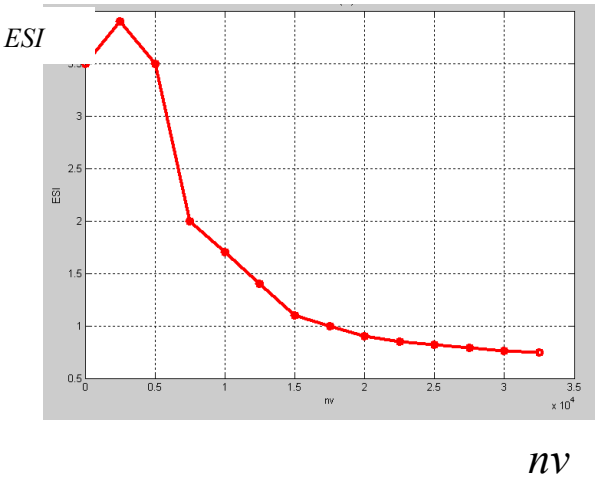


Рис. 3.15. Зміна ентропії системи залежно від довжини переданих мережних пакетів $ESI = f(nv)$

Для виявлення КНІ застосовувався метод ковзного вікна [5]. Атака 1-*ої* групи була виявлена менш ніж за 1 с. Атаку 2-*ої* групи виявити не вдалося. Отже, для виявлення таких КНІ необхідно шукати більш вагомі описові ознаки $\{p_{ax1},\ldots,p_{axn}\}$, які дозволять усунути цей недолік.

У цілому результати моделювання свідчать про працездатність розглянутого підходу й можливості розпізнавання станів ІКС та АСК за прийнятний час і, практично, з гарантованою надійністю, під якою розуміємо нижній поріг числа помилок розпізнавання.

Отже, можна припустити, що порівнянням кривих для випадків “типового” (відсутність атаки) і змішаного (наявність атаки на тлі “типового”) трафіків можна виявити наявність атаки. Таким чином, порівнювані залежності можуть бути додатковою ознакою для ДПРЗ ІБ.

Оскільки кожний стан системи може характеризуватися сукупністю значень квантованих цифрових сигналів, характерних для S_i, то, у термінах ДПРЗ ІБ (див. розділ 2), число градацій ознаки – рівня квантування, виступає як універсальна множина, потужність якого дорівнює максимальному рівню квантування, характерному для даної моделі.

Виконані дослідження дали змогу розробити наступну послідовність етапів моделювання роботи СЗІ у складі ІКС та АСК при впливі неоднорідних потоків даних (див. табл. 3.3).

Таблиця 3.3

Послідовність етапів моделювання ІБ ІКС та АСК

№ етапу	Опис послідовності дій при моделюванні станів системи	Джерело
1	побудувати схему багатофазної СМО для розрахунків ємнісних і часових характеристик СЗІ ІКС	див. розділ 3.2
2	провести аналіз функціонування розглянутого ЗЗІ з урахуванням даних розрахунків ємнісних і часових характеристик СЗІ ІКС	

Продовження табл. 3.3

3	зробити декомпозицію системи	Графи переходів, див. рис. 3.9, 3.11–3.16
4	виконати формалізацію всіх можливих станів модельованої системи й окремих вузлів	
5	виділити небезпечні стани ІКС	
6	визначити інтенсивності й ймовірності переходу вузлів системи зі стану в стан	
7	визначити ймовірнісні характеристики станів системи як функції часу й інтенсивності вхідних потоків вимог	див. формули 3.1–3.2, 3.11–3.27
8	скласти для кожного вузла системи повну групу стану системи – $\{A\Omega_1,...,A\Omega_N\}$, яким будуть відповідати первинні специфікації	
9	визначити оцінки розподілу ймовірностей P_{S_i}, характерних для станів системи, у які вона потрапила в результаті КНІ	
10	визначити зміни етропій усіх підсистем ESI_{S^*}	див. формулу 3.38
11	сформувати випадкову стохастичну мультимножину $M\Theta_L^*$	див. формулу 3.39
12	обчислити інформаційні відстані між мультимножинами	
13	вибрати окремі рішення на користь стану, для якого величина $DIS(M\Theta_i, M\Theta_L^*)$ є найменшою для кожної ознаки p_{axi}. Обчислити вагові коефіцієнти окремих рішень	див. формулу 3.40
14	обираємо відповідно до процедури «голосування», описаній в розділі 2, стан системи з більшим ваговим коефіцієнтом	див. формулу 3.41

Порівняльний аналіз існуючих підходів до моделювання КНІ у ІКС представлений у вигляді таблиці 3.4, надає можливість зробити висновок про необхідність впровадження в сучасних умовах розробленої моделі та системи підтримки прийняття рішень з визначення фактів КНІ у ІКС та АСК.

Таблиця 3.4

Порівняльний аналіз методів моделювання КНІ у ІС

	Логічні		Методи засновані на графах					Комбіновані
	Логіка 1-го порядку	Нечітка логіка	Графи атак	Графи з кінцевою множиною станів	Ланцюги Маркова	Байєсовські мережі	Мережі Петрі	Ланцюги Маркова + нечітка логіка для розпізнавання атаки
точність	+	-	-	-	-	+/-	+/-	-
можливість пояснення	+/-	-	-	-	-	+	-	-
швидкодія	-	+/-	+/-	+/-	+	+/-	-	-
здатність до навчання	+	-	+/-	+/-	+	-	+/-	-
«+» – наявність труднощів; «-» – відсутність труднощів; «+/-» – часткові труднощі								

Варто зазначити, що саме по собі дослідження можливих станів системи, у тому числі тих, у яких вона може перебувати в результаті дій зловмисника (навіть з урахуванням фактора використання класичних ЗЗІ), не гарантує того, що дана система дійсно виявиться ефективною в умовах виникнення нових загроз і протидії інформаційним ризикам. Тому в наступному розділі роботи ми детально зупинимося на питанні оптимізації комплексів ЗЗІ ІКС, /що разом із результатами досліджень, викладеними в 2-му та 3-му розділах посібника, дозволить виробити конкретні рекомендації з удосконалювання методології побудови захищених ІКС та АСК.

3.4. Висновки до розділу 3

У результаті проведених у 3-му розділі навчального посібника, можна зробити наступні висновки.

1. Встановлено, що Марковські моделі процесів широко використовуються при аналізі й синтезі СЗІ ІКС, причому властивість марковості є певним обмеженням на використовувані реальні сигнали, але цілком достатнім для розробки змістовних методів аналізу й синтезу комплексів СЗІ.

2. Визначено, що математичні моделі із використанням апарату ланцюгів Маркова й мереж Петрі є ефективним інструментом для кількісної оцінки можливості реалізації КНІ у ІКС.

3. Запропоновано математичну модель функціонування СЗІ ІКС при неоднорідних потоках вимог та мережних класах загроз. В моделі апріорі виділяються найбільш інтенсивні вхідні потоки і враховується наявність черг за потоками, що вимагають швидкого обслуговування в апаратній частині комплексів СЗІ ІКС.

4. Удосконалено моделі інтелектуального розпізнавання загроз ІКС, з урахуванням можливостей зміни нападаючими інтенсивності неоднорідних потоків запитів. Досліджено динаміку станів СЗІ ІКС у разі декількох неоднорідних потоків вимог при різних типах нападу на інформацію у ІКС. Ймовірність вирішення завдання розпізнавання загрози нападу на інформацію та створення небезпеки для ІКС склала 85–98%, залежно від типу КНІ на ресурси ІКС або АСК.

5. Розроблено послідовність етапів моделювання роботи комплексів СЗІ ІКС при впливі неоднорідних потоків даних.

6. Для подальшого вирішення завдань навчального посібника та з метою перевірки отриманих у розділах 2 та 3 результатів, потрібно провести дослідження з імітаційного моделювання інтелектуального розпізнавання загроз нападу на інформацію та її захисту у ІКС.

РОЗДІЛ 4
ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ КОМПОНЕНТІВ БЕЗПЕКИ ІНФОРМАЦІЙНО - КОМУНІКАЦІЙНОГО СЕРЕДОВИЩА

4.1. Формалізація задачі імітаційного моделювання нападу на інформацію та її захисту у інформаційно - комунікаційному середовищі

Великі системи, у тому числі АСК та ІКС, оснащені комплексами ЗЗІ, складаються з сотень, а в деяких випадках із тисяч елементів і ще більшої кількості зв'язків між ними. Такі системи характеризуються неоднорідністю елементів і неоднорідністю зв'язків. Незважаючи на те, що окремі елементи або зв'язки прекрасно описуються моделями дискретної математики або теорії масового обслуговування (див. розділи 2, 3), про систему в цілому цього сказати не можна. Природною альтернативою є використання імітаційного моделювання, яке дозволяє поєднати між собою різнорідні математичні моделі елементів, що входять до складу ІКС. Імітаційне моделювання є одним із методів, які дозволяють оцінити ЗЗІ ІКС та її реакцію на спроби НСД (збурення) за рядом показників [1].

Складні системи, до яких належить і ІКС, відрізняються властивостями, що можуть стати причиною виникнення багатьох похибок при спробі поліпшення поведінки системи. До них належать:

Мінливість. Характеристики ІКС постійно змінюються тому, що у процесі розвитку АПК перетворюються характеристики елементів СЗІ.

Наявність зовнішнього середовища. Кожна АСК або ІС існує у зовнішньому оточенні і за своєю суттю є підсистемою більш масштабної ІКС. Зовнішнє оточення ІКС становить комплекс елементів із визначеними властивостями, які при їх модифікації можуть викликати зміну стану ІКС. Тому зовнішнє оточення системи повинно бути описане усіма зовнішніми факторами, які можуть спричиняти вплив на систему.

Тенденція до погіршення характеристик. Характеристики ІБ складних систем, як правило, із часом погіршуються. Це повною мірою стосується і ІКС.

Взаємозалежність. Кожна подія з ІБ у складній ІКС залежить від попередніх подій і впливає на наступні. Крім того, різноманітні процеси у реальних умовах реалізують паралельно і у підсумку, створюють вплив один на одного.

Організація. Складні системи складаються з елементів, що характеризуються високим ступенем організації. Елементи об'єднуються у ієрархії підсистем, які взаємодіють між собою для виконання цільового призначення системи. Доцільно зауважити, що вибір елементів СЗІ, які вводяться або виводяться із системи ІБ, та їх конфігурація визначаються дослідником.

При імітаційному моделюванні доцільне використання схематичних моделей, які за своєю суттю є відображенням дійсного перебігу подій, що сприяє поглибленому розумінню процесу функціонування системи.

Схема етапів створення імітаційних моделей компонентів СЗІ ІКС та взаємозв'язки валідації, верифікації і встановлення довіри представлено на рис. 4.1.

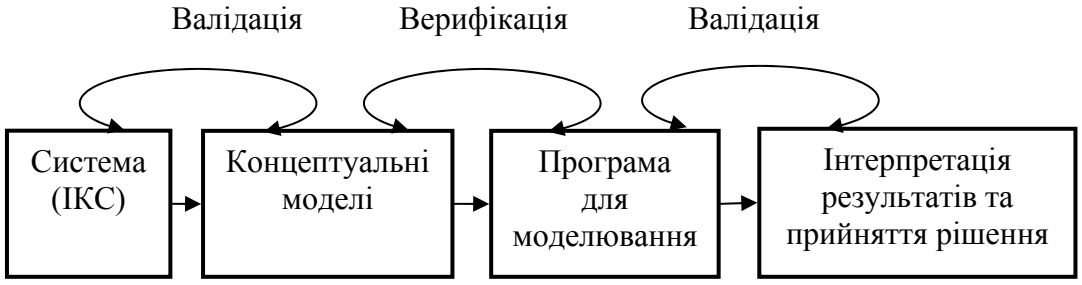


Рис. 4.1. Схема етапів створення імітаційний моделей компонентів СЗІ ІКС

Валідацію можна протиставити етапу інтерпретації або аналізу вихідних даних моделювання, який становить статистичну задачу, що пов'язана з оцінкою достовірності результатів, отриманих за допомогою імітаційної моделі СЗІ ІКС.

Кінцева перевірка адекватності імітаційної моделі підтверджується за умови, що її вихідні дані ідентичні вихідним даним реальної СЗІ. Якщо система,

аналогічна запровадженій, існує у поточний момент часу, то розробляють імітаційну модель і порівнюють вихідні дані. У тому випадку, коли два комплекти даних виявляються подібними, модель системи рахується адекватною.

За допомогою імітаційного моделювання при створенні ЗЗІ можуть вирішуватися такі завдання: визначення шляхів удосконалення ІКС та АСК на підставі аналізу різних варіантів технічної, технологічної, а також організаційної перебудови та дослідження наслідків прийнятих рішень. Імітаційне моделювання дозволяє відпрацьовувати не тільки різні варіанти структур і режимів функціонування технічних засобів і програмного забезпечення, але і різних форм функціонування ЗЗІ ІКС.

Ключовим рішенням при побудові моделюючого засобу ІБ структурних підрозділів об’єкту інформаційної безпеки є вибір середовища для створення моделей, оскільки від цього залежатимуть можливості проведення експериментів, представлення результатів і доопрацювання моделей. На етапі побудові моделі виникає питання про вибір інструментарію. При цьому важливими якостями для створення ефективної моделі є:

детальна реалізація протоколів, задіяних в КНІ;

можливість написання і підключення власних модулів для реалізації ДПРЗ (інтелектуального розпізнавання загроз КНІ);

можливість зміни параметрів моделювання під час проведення експериментів;

платформна незалежність;

розвинений графічний інтерфейс.

4.2. Імітаційні моделі в MATLAB та Simulink інтелектуального розпізнавання загроз при нападах на інформацію та її захисту у інформаційно-комунікаційному середовищі

Для імітаційного моделювання роботи ЗЗІ ІКС був обраний пакет MATLAB 7/Simulink (Ліцензія 62228-07509-32393-00528) [5].

Загальна схема моделювання, яка може бути затребувана фахівцями у сфері ІБ компанії, у тому числі тих, що працюють у територіально віддалених ЛВ, показана на рис. 4.2.

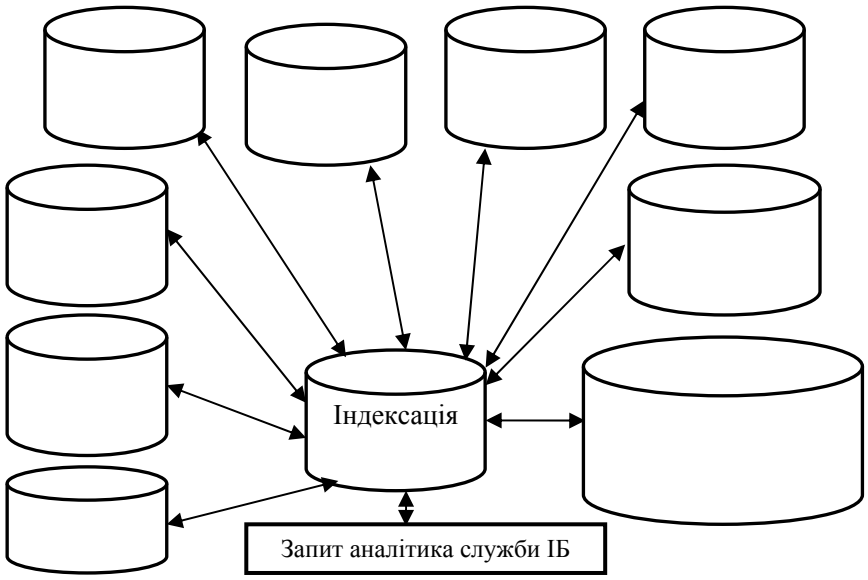


Рис. 4.2. Фізична структура моделювання ІБ ІКС

Всі математичні моделі, описані в розділах 2,3, були реалізовані в пакеті MATLAB 7 для подальшого дослідження режимів роботи структурних підрозділів об’єкту інформаційної безпеки в умовах протидії різних варіантів КНІ у ІКС, а також для перевірки адекватності отриманих моделей.

Відповідні імітаційні моделі розроблені для найбільш поширених та уразливих до нападу на інформацію компонентів ІКС – комутаторів, міжмережевих екранів, систем відеоспостереження, АРМ, МАРМ, серверів, PLC системи керування рухом та ін.

Для реалізації процесу нечіткого моделювання процедури інтелектуального розпізнавання по окремих класах загроз ІКС, та на основі таблиць із базами знань (див. табл. 2.4, 2.5, 2.7) в середовищі MATLAB (пакет розширення Fuzzy Logic Toolbox) були складені відповідні правила для експертної системи, див. рис. 4.3.

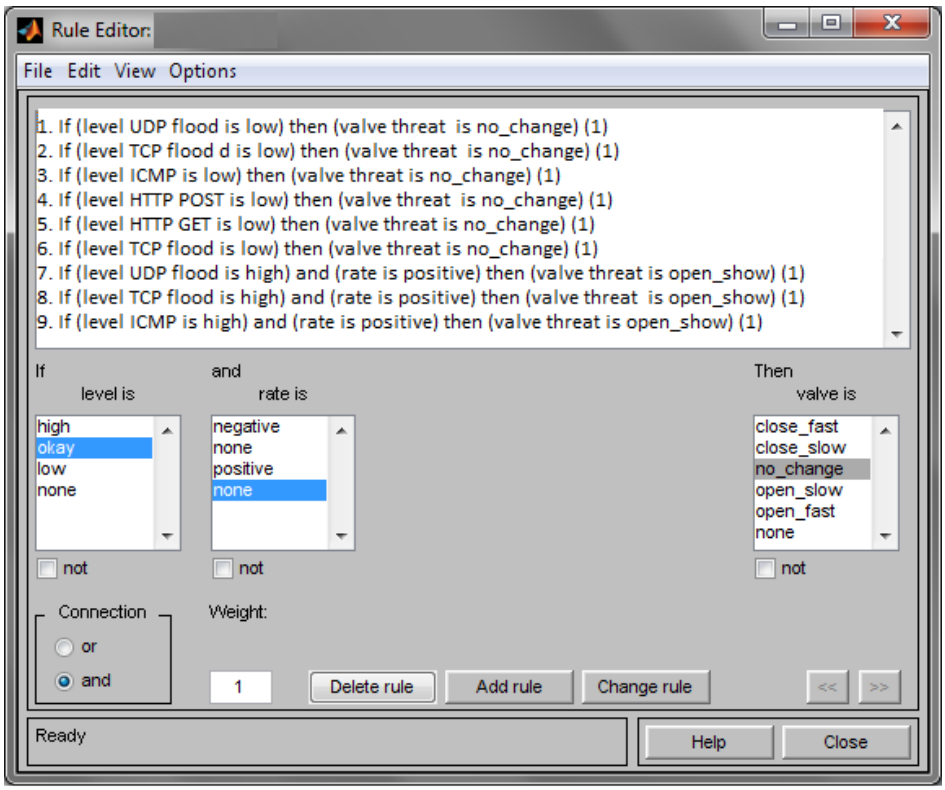


Рис. 4.3. Система правил для ЕС розпізнавання загроз ІКС

Враховуючи все вищесказане, також використовуючи отримані вирази (3.9 – 3.37), було виконано імітаційне моделювання різних режимів функціонування ІКС та АСК зі змінною структурою й неоднорідних потоками даних [10].

Отримані вирази (3.9 – 3.37) були реалізовані в середовищі Matlab 7 (у вигляді m-*файлів*) і Simulink (*.mdl) [5 – 4].

На рис. 4.4 – 4.6, 4.8, 4.13, 4.15, 4.31, 4.33, 4,34 показані основні схеми для імітаційного моделювання режимів роботи ІКС, АКС в умовах впливу різних класів КНІ на сервер, АРМ, АСК рухом та ін.

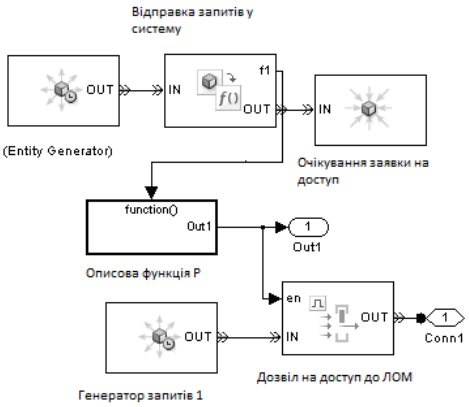


Рис. 4.4. Підсистема моделювання потоків запитів до серверів ІКС

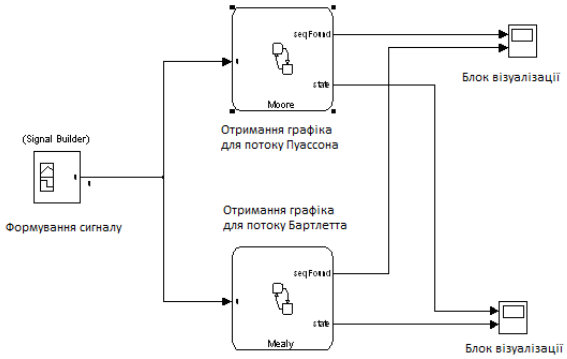


Рис. 4.5. Стани системи для неоднорідних потоків вимог

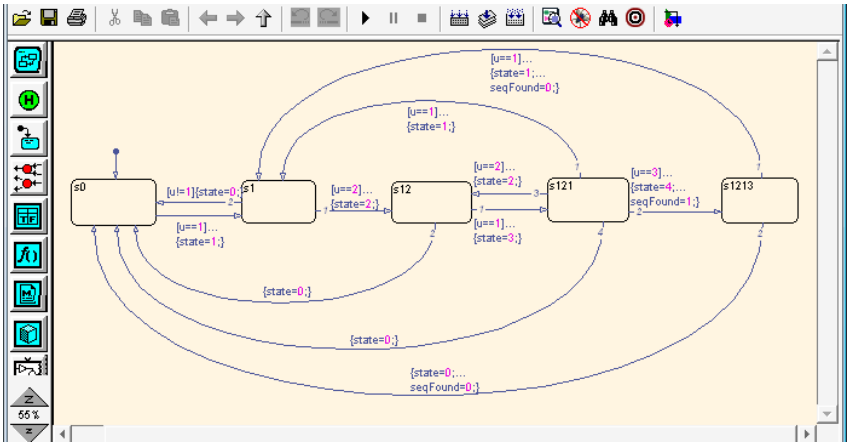
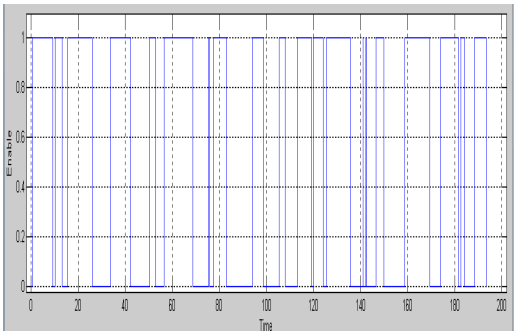


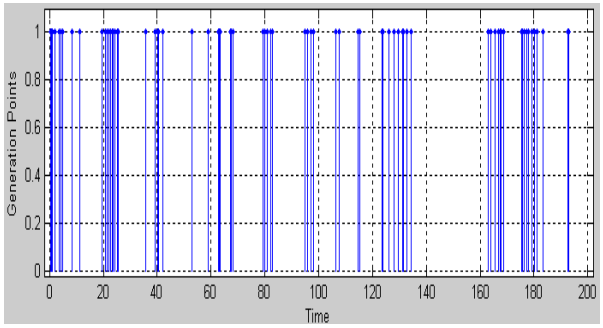
Рис. 4.6. Опис зміни станів системи при вхідному потоці окремих вимог

Для всіх трьох змодельованих варіантів потоку заявок (k_1, k_2, k_3) на обслуговування задаємо параметри для генератора запитів (у вигляді векторів вхідних потоків). У ході моделювання ми програмно змінювали кількість заявок на обслуговування в діапазоні від 10 до 100000 і, відповідно, режими (закон) розподілу надходження заявок – по потоку Пуассона (звичайний режим роботи ІКС) і потоку Бартлетта (неоднорідних потік заявок, створений у результаті КНІ).

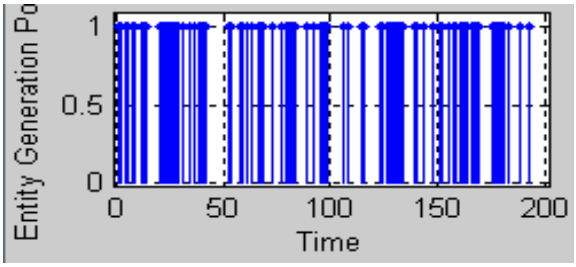
Отримані в ході досліджень залежності показані на рис. 4.7.



а)



б)



в)

а) розподіл потоку заявок за показовим законом (потоки Пуассона) $10 < k_1 < 100$;

б) розподіл потоку заявок за показовим законом (потоки Пуассона) $10 < k_1 < 100$ і $100 < k_2 < 200$;

в) розподіл неоднорідного потоку (потік Бартлетта) $10 < k_1 < 100$ і $100 < k_2 < 200$ й $1000 < k_3 < 100000$

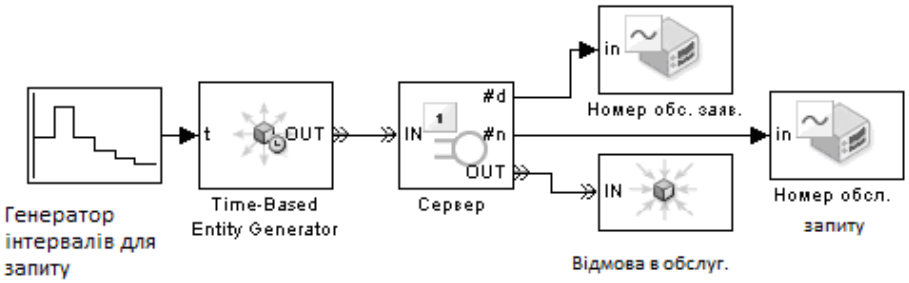
Рис. 4.7. Залежність розподілу точок доступу заявки на обслуговування від часу й довжини черги

Як бачимо із графіків (див. рис. 4.7 а) б)), при звичайних режимах функціонування серверів ІКС або АСК, тобто при показовому законі розподілу потоку заявок (потоки Пуассона), наприклад в модулях e-business або системі зв'язку на з/т GSM-R, час обслуговування заявки варіюється в припустимих межах і черга не виникає.

Однак, якщо в результаті впливу на чергу в ній виділяються пріоритетні потоки (див. рис. 4.7 в), то ситуація докорінно змінюється, Наприклад, зловмисникові, що увійшов у систему, вдалося створити кілька інтенсивних вхідних потоків. Тоді, по-перше, неможливо підсумувати деякі потоки заявок і

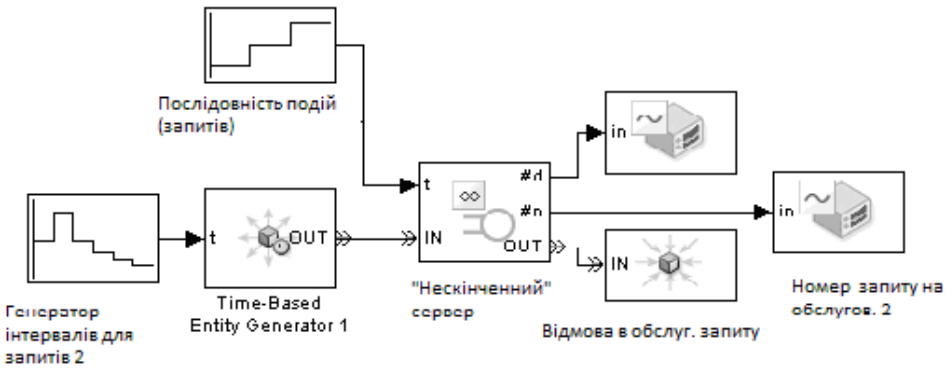
звести завдання до одномірного випадку, по-друге, обслуговування заявок неоднорідних потоків здійснюється в інтервали часу, що не перетинаються, по-третє, існують інтервали неприступності, протягом яких потоки не обслуговуються внаслідок використання СЗІ.

У ході моделювання розглядалися схеми обслуговування заявок в ІКС, коли з метою підвищення надійності роботи системи в складі ЛОМ функціонує не один рис. 4.8 а), а відразу кілька серверів (див. рис. 4.8 б), в)).



а)

б)



в)

а) схема з одиничним сервером; б) схема з декількома серверами;
в) схема з «нескінченним» сервером (Infinite Server)

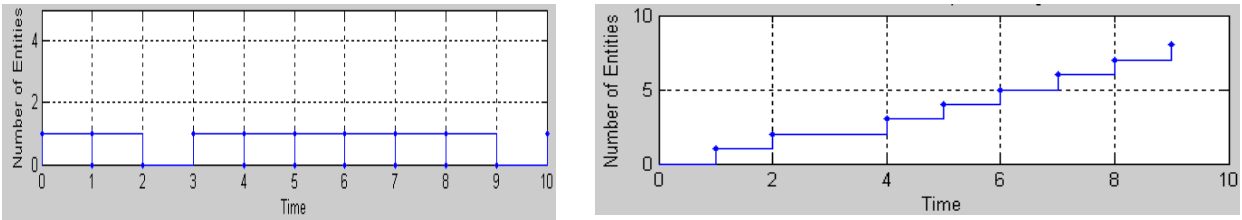
Рис. 4.8. Схеми обслуговування заявок в ІКС

На рис. 4.9 – 4.12 показані основні результати моделювання потоків заявок k_1, k_2, k_3 в ІКС.

У межах роботи розглядалися в основному варіанти КНІ на мережу ІКС та АСК рухом типу «відмова в обслуговуванні» (DoS/ DDoS), як найбільш поширених варіантів та ймовірних атак на ІКС [10]. Як відомо, побічним ефектом таких атак є великий трафік, спрямований на атакований ресурс АРМ, МАРМ, модулів e-business ІКС, що часто ігнорується мережними адміністраторами і вважається звичайною поведінкою мережного сегмента атакованого сервера.

У ході імітаційного моделювання розглядалися основні КНІ DoS/DDoS: блокування каналу зв'язки; блокування мережного сервісу.

Як і варто було сподіватися, при збільшенні кількості серверів або використанні сервера з необмеженою довжиною черги, кількість обслужених заявок зростає і застосування різних типів вхідних потоків на процес не впливає. Але цілком очевидно, що дозволити собі відразу кілька серверів може далеко не будь-яка компанія, і тут необхідно розрахувати співвідношення ризиків, пов'язаних із втратою інформації, та витрат на її захист, тобто провести дослідження з оптимізації комплексів СЗІ ІКС.

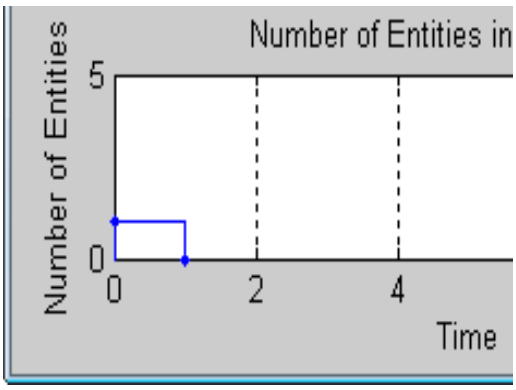


а) б)

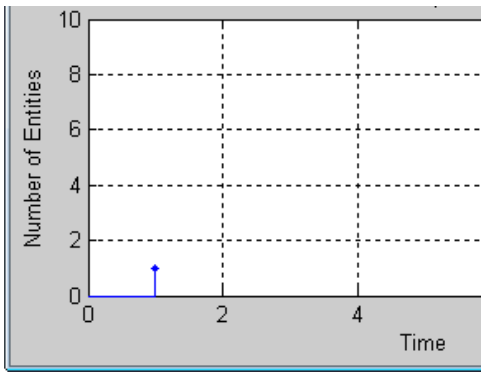
а) розподіл потоку заявок $10 < k_1 < 100$

б) кількість заявок, що обслуговуються, при $100 < k_1 < 200$

Рис. 4.9. Розподіл потоку заявок, що обслуговуються,
при використанні одного сервера та вхідному потоці вимог Пуассона



а)

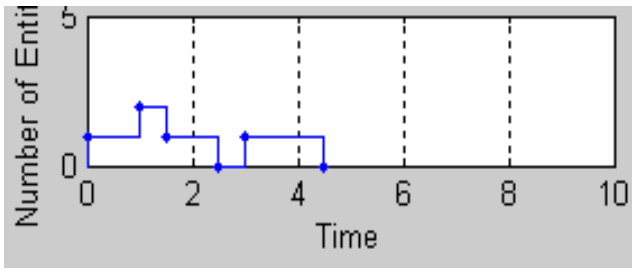


б)

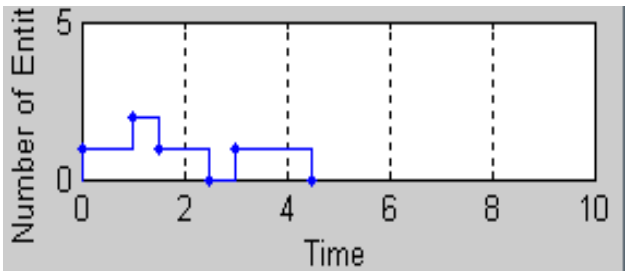
а) розподіл потоку заявок при $10 < k_1 < 100$, $100 < k_2 < 200$ і $1000 < k_3 < 100000$

б) кількість заявок, що обслуговуються, $100 < k_1 < 200$, $200 < k_2 < 500$ і $10000 < k_3 < 50000$

Рис. 4.10. Розподіл потоку заявок, що обслуговуються, при створенні неоднорідного потоку (потік Бартлетта) для одного сервера



а)

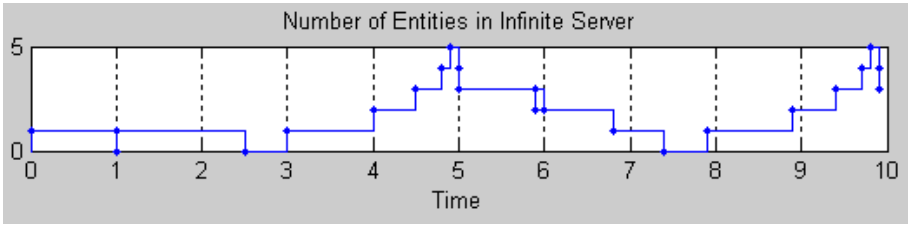


б)

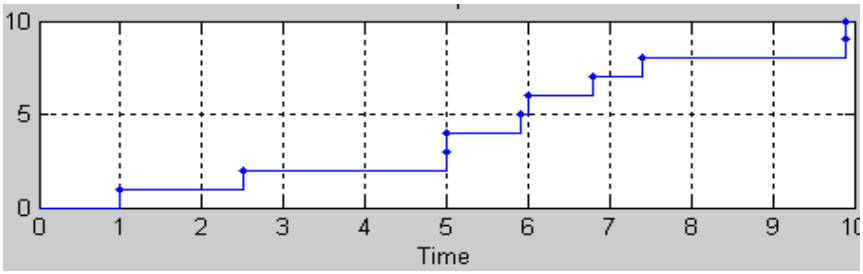
а) розподіл потоку заявок при $10 < k_1 < 100$, $100 < k_2 < 200$ і $1000 < k_3 < 100000$

б) кількість заявок, що обслуговуються, $100 < k_1 < 200$, $200 < k_2 < 500$ і $10000 < k_3 < 50000$

Рис. 4.11. Розподіл потоку заявок, що обслуговуються, при створенні неоднорідних потоку (потік Бартлетта) для двох серверів



а)



б)

а) розподіл потоку заявок при $10 < k_1 < 100$, $100 < k_2 < 200$ і $1000 < k_3 < 100000$

б) кількість заявок, що обслуговуються, $10 < k_1 < 100$, $100 < k_2 < 200$ і $1000 < k_3 < 100000$

Рис. 4.12. Розподіл потоку заявок, що обслуговуються, при створенні неоднорідних потоку (потік Бартлетта) для «нескінченного» сервера

Для дослідження можливості виявлення КНІ DoS/DDoS був проведений імітаційний експеримент у системах, що представляє собою сегмент комп'ютерної мережі підприємства. При цьому мережа працювала у звичайному режимі й зазнала впливу атаки.

Для візуалізації сигналів був спроектований спеціальний блок – «Signal Visualization», який дозволяє аналізувати основні параметри ЛОМ на рівні переданих пакетів даних [4-15].

За допомогою генератора трафіка створювалися атаки 1-*ої* і 2-*ої* групи.

Лістинги основних функцій, що описують сегмент ІКС, у вигляді *m* файлу наведені в додатку Б.

Враховуючи рекурентні вирази (3.11–3.27) і використовуючи інструментарій імітаційного моделювання пакету MATLAB 7 та Simulink, розглянемо один із підходів до аналізу динаміки і взаємозв'язку трафіків у

модельній системі ІКС з урахуванням неоднорідних потоків даних. Для прикладу розрахунку тестових трафіків у цій роботі припускалося, що мережа ІКС складається з однієї лінії передачі даних і трьох станцій (АРМ), які періодично надсилають вимоги на передачу даних по лінії (див. рис. 3.18). Параметри запитів були сформовані відповідно до даних розділу 3 (АРМ1 – малоінтенсивний пріоритетний потік k_1; АРМ2 – малоінтенсивний потік k_2; АРМ3 – пріоритетний потік найбільшої інтенсивності k_3), див. рис. 4.13.

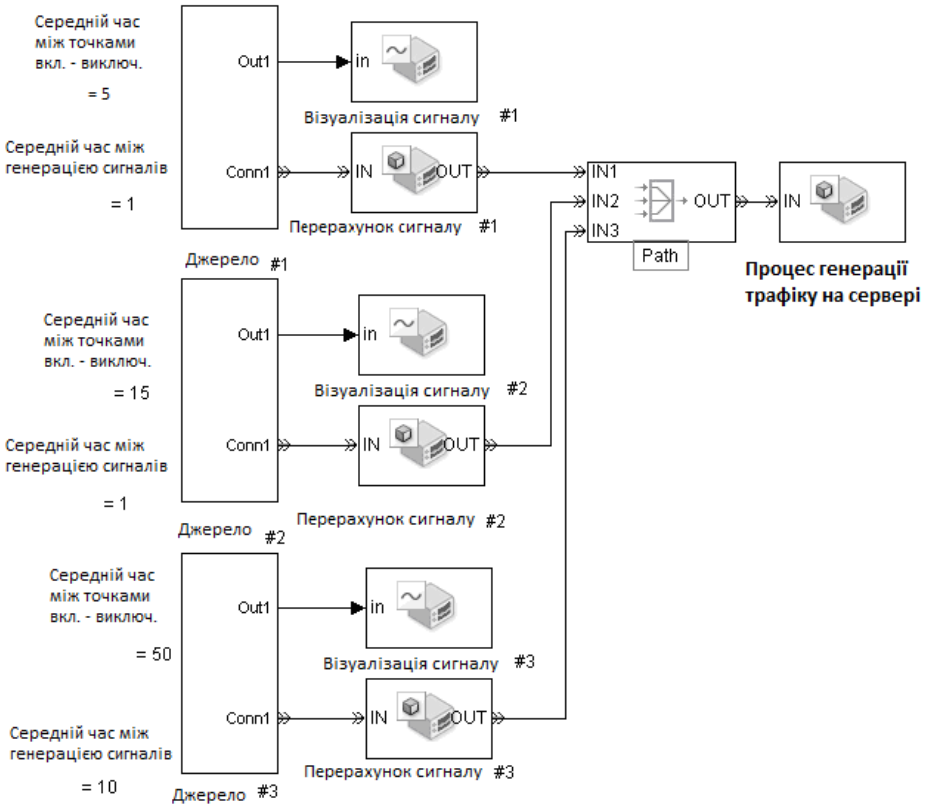


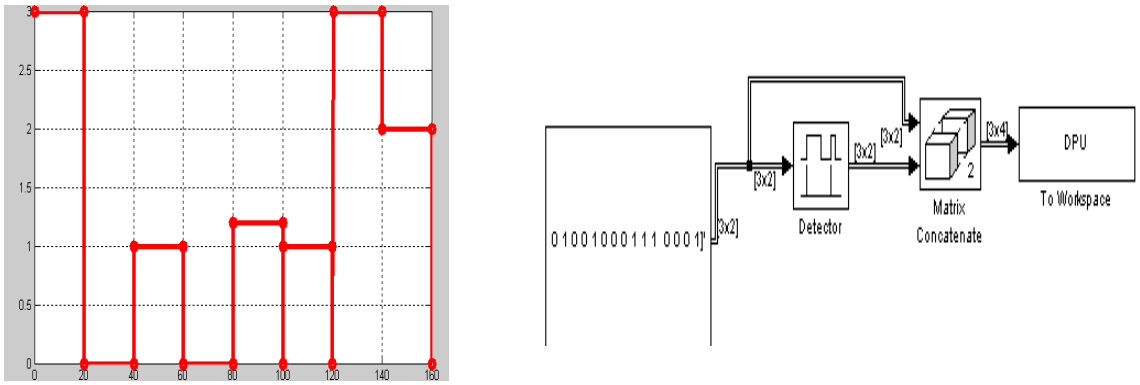
Рис. 4.13. Схема імітаційного моделювання ІКС

Також припускалося, що час дискретний і змінюється від 0 до деякого значення *T*. АРМ працюють незалежно один від одного, і в кожний момент часу з певною ймовірністю від будь-якої станції може надійти вимога на передачу даних по лінії або відбутися звільнення лінії.

У блоці аналізу трафіку, використовуючи ДПРЗ ІБ, можна будувати розподілені системи виявлення та блокування DoS/DDoS – атак, несанкціо-

нованої мережевої активності, атак на мережеві служби, документування мережевих повідомлень, а також інших завдань інформаційної безпеки ІКС.

Програмна реалізація ММ дозволяє отримати наочне уявлення про процеси на виході кожної станції і на загальній лінії – визначити важливі характеристики СМО (час простою лінії, середній час передачі даних по лінії від кожної станції і т. п.). Практично нескладно при необхідних значеннях кількості станцій побудувати програмно тестові послідовності будь-яких обраних користувачем або найбільш важливих компонент стану ІКС. Наприклад, на рис. 4.14 а), зображений тестовий трафік на лінії, коли у всіх станцій однакові ймовірності заняття лінії і приблизно однакові ймовірності звільнення лінії після її заняття станцією. Для інших випадків, що часто зустрічаються (наприклад, перша і друга станції мають більше запитів на передачу даних і довше займають лінію, ніж перша станція) бажано мати свій набір тестових трафіків. На рис. 4.14 б показаний модуль детектора для виявлення аномальності квантового сигналу.



а) б)

Рис. 4.14. Фрагмент тестової послідовності на лінії (для випадку, коли для всіх станцій однакові ймовірності звернення до сервера і зняття запиту)

Даний підхід може бути використаний при аналізі роботи та діагностиці СМО, що складаються з будь-якої кількості станцій. У загальному випадку

станції (АРМ) можуть бути і залежними одна від одної. При цьому збільшиться число станів системи, ускладниться діаграма переходів і зміниться розмірність ймовірнісної матриці. Даний спосіб вибору і задавання параметрів СМО гарантує відповідні ймовірнісні характеристики реалізацій процесів у всіх характерних точках системи (див. рис. 4.15 – 4.18).

В силу «аналогового» ММ (задається ймовірностями переходів і т. д.) для отримання тестових трафіків варто обирати параметри ММ СМО близькі до реально існуючих на підставі апріорної інформації про роботу станцій, коли малі локальні відхилення від еталонних трафіків можуть призводити до однозначно розпізнаваних сигнатур.

Згідно з нашими викладками, наведеними в розділі 3, розглянемо, як змінюється ефективність DoS/DDoS – атаки при великий кількісті потоків і для ситуації, коли потоки мають різні параметри. На рис. 4.19 показана модель черги, складена для різних типів потоків.

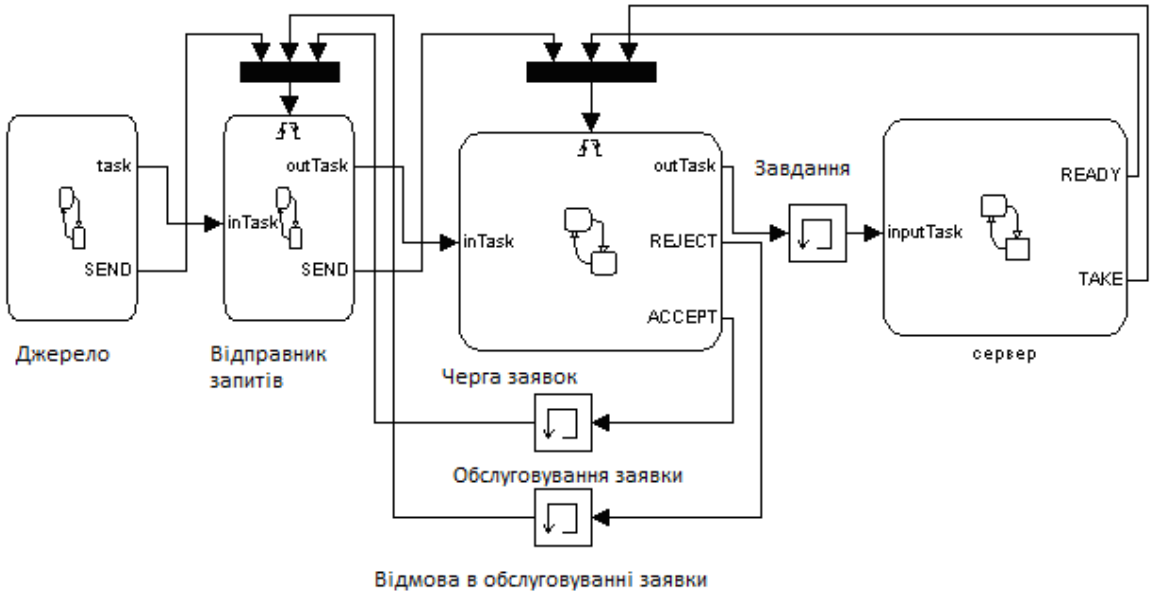


Рис. 4.15. Схема функціонування сервера ІКС, що обслуговує заявки

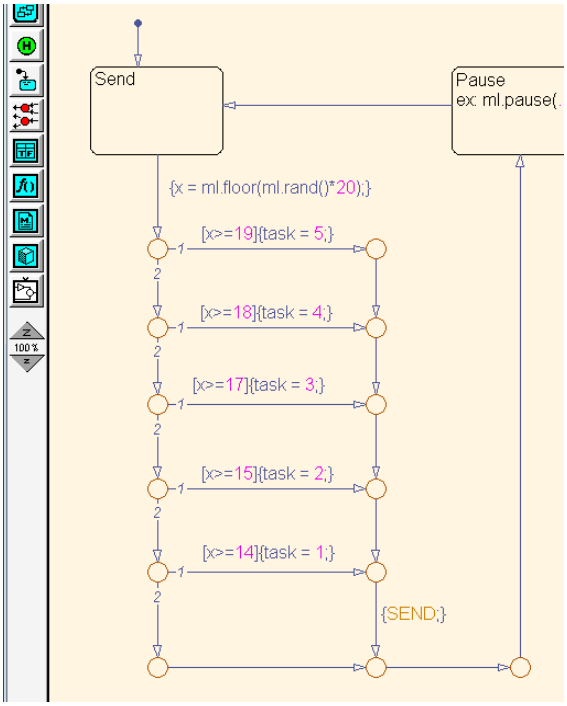


Рис. 4.16. Блок «Джерела запитів»

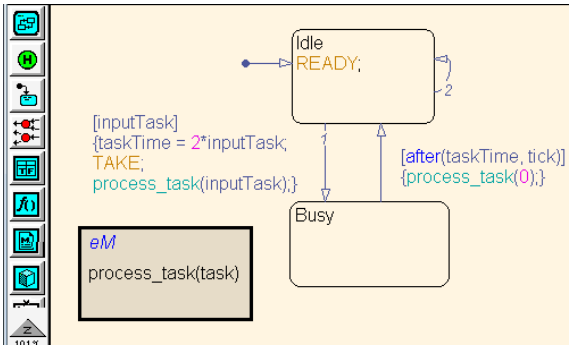


Рис. 4.17. Блок «Сервер»

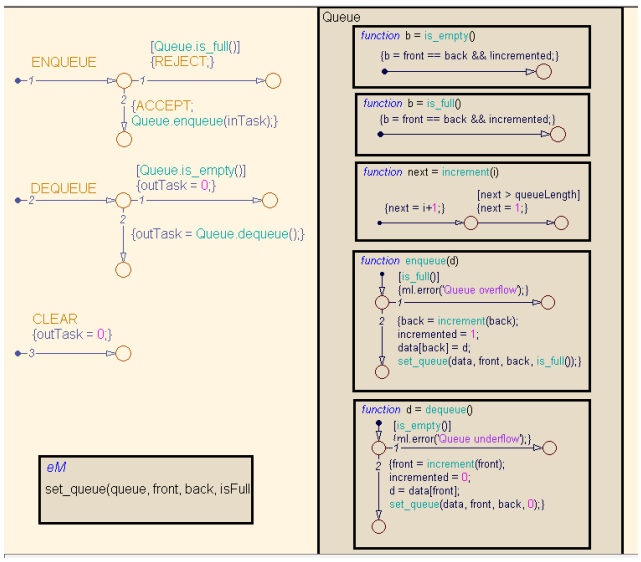


Рис. 4.18. Блок «Відправник»

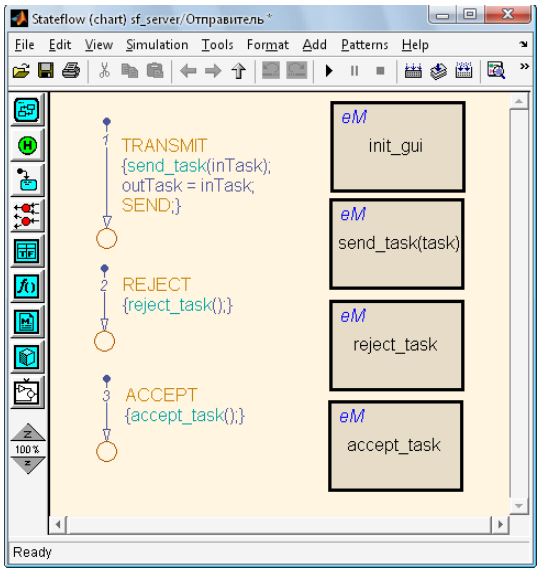


Рис. 4.19. Блок «Черга заявок»

Для візуалізації потоків заявок використовувався компонент Scope. На рис. 4.20 і 4.21 показані результати моделювання роботи сервера ІКС при звичайному пуассонівському потоці (рис. 4.20) і неоднорідному (конфліктному) потоці, з виділеним пріоритетом на виконання за двома потоками заявок (рис. 4.21).

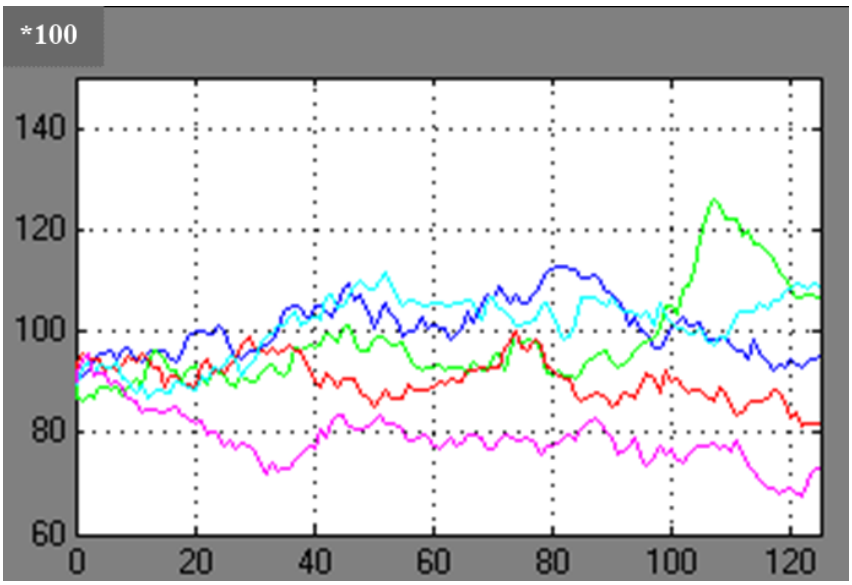


Рис. 4.20. Візуалізація потоків заявок до сервера (для пуассонівських потоків запитів)

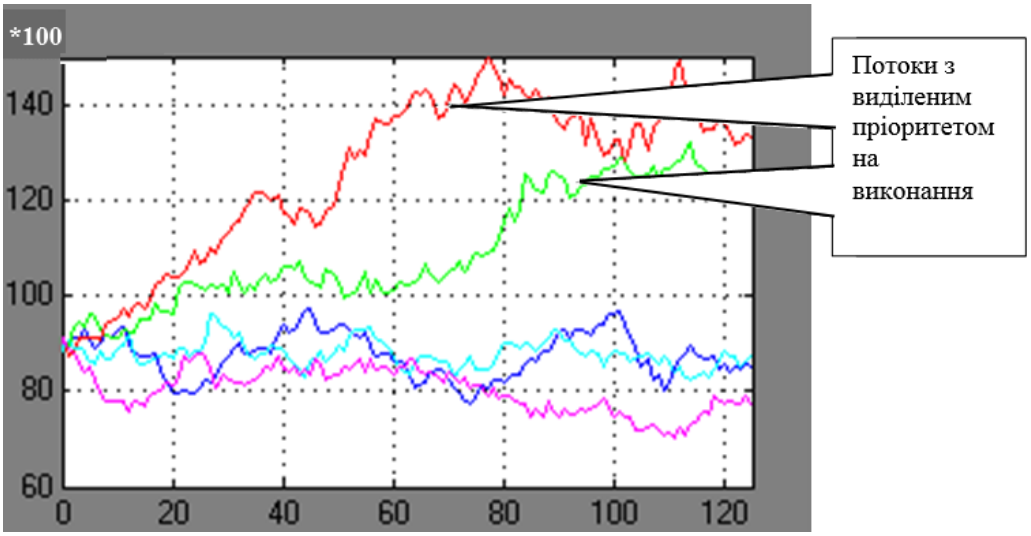
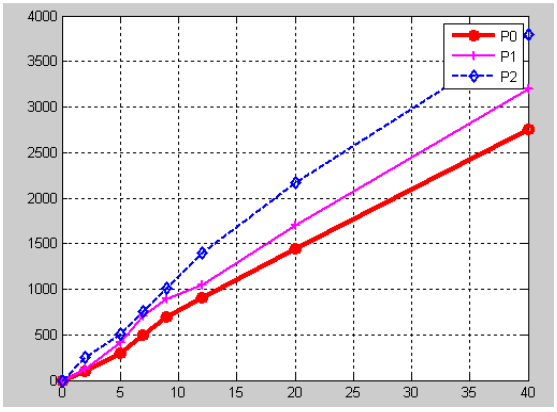


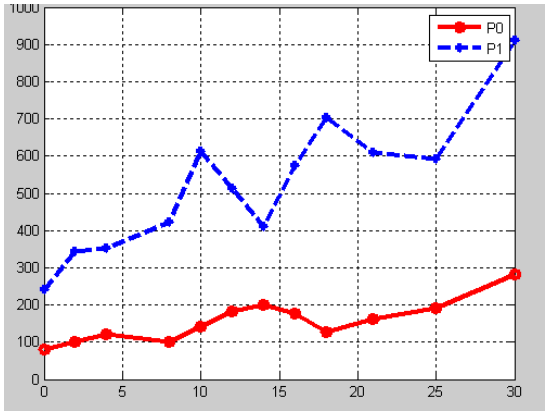
Рис. 4.21. Візуалізація потоків заявок до сервера для конфліктних потоків

Для різних варіантів атак на ІКС або АСК розглядалися додаткові (неоднорідні) потоки, сформовані зараженим абонентом на інші АРМ, які в даний момент часу перебувають на зв'язку із сервером або вихідним терміналом. На рис. 4.22 – 4.23 показані результати моделювання.



P0- без атаки; P1, P2- атаки DoS

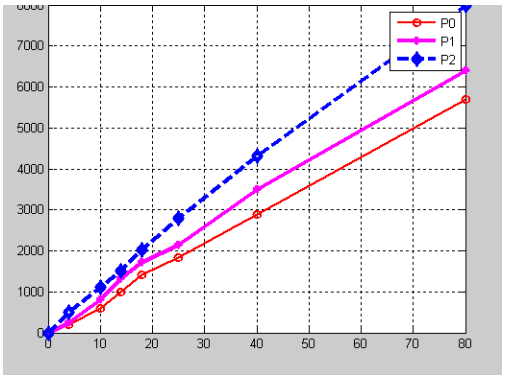
а) сумарний потік заявок



P0- без атаки; P1 – атака DoS

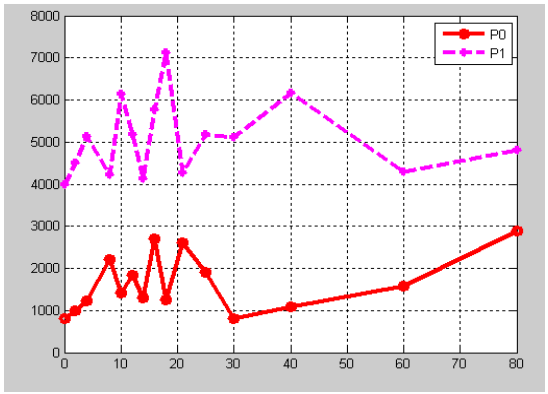
б) середній потік заявок

Рис. 4.22. Розподіл сумарного й середнього пуассонівського потоку заявок



P0- без атаки; P1, P2- атаки DoS

а) сумарний потік заявок



P0- без атаки; P1 – атака DoS

б) середній потік заявок

Рис. 4.23. Розподіл сумарного й середнього потоку заявок при створенні неоднорідного потоку (потік Бартлетта)

З отриманих графіків можна зробити висновок про закономірності можливого збільшення переданого по ЛОМ середнього й сумарного потоку за рахунок додавання конфліктних складових до потоку опрацьованих до ІКС запитів.

В ході імітаційного моделювання ми розглядали сценарій, при якому аналізувався вплив DoS (DDoS) – атаки на «частотну характеристику». Крива,

позначена на рис. 4.24, є залежністю нормалізованої пропускної здатності модулів ІКС та АСК з множиною потоків трафіку від часу.

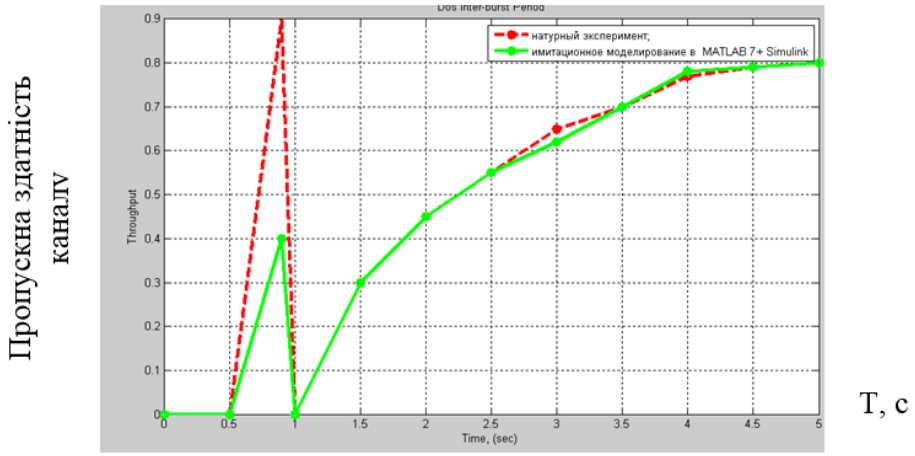


Рис. 4.24. Пропускна здатність при атаці – моделювання та експеримент

Варто врахувати, що середня швидкість атаки зменшується зі збільшенням часу. Як бачимо з графіка 4.23, ефективність атаки абсолютно не зростає зі збільшенням середньої швидкості. Для нас найбільш цікавими є ділянки, на яких пропускна здатність стає близькою до нуля. Тобто, якщо зловмисник створює збій в каналі, відправляючи пакети з мінімальною періодичністю, то він повністю блокує трафік TCP. Як тільки відбувається короткочасне блокування каналу, всі потоки трафіку, що йдуть через нього, зупиняються. При цьому найкращий момент для наступної посилки атакуючого – $T = min$ RTO (RTO – часова шкала з позначками тайм-аутів повторної передачі пакетів [4-15]). У разі, якщо $T>min$RTO, період атаки збільшується, а це веде за собою збільшення загальної пропускної здатності каналу в перервах між посилками атакуючого.

В ході моделювання як основна топологія атакованої мережі підприємства були прийняті такі параметри:

кілька потоків ідуть через вузький перевантажений модуль ІКС ємністю до 1.5 Мб/с;

розмір буфера черги обраний таким чином, що RTT (шкала оцінки часу проходження пакетів по каналу зв'язку до адресата і назад) лежить у межах від 12 до 132 мс.

поодинокі заявки надходять (імпульси DoS – трафіку) зі швидкістю 1.5 Мб/с, тривалістю імпульсу 100 мс і розмір пакета 50 байт.

На рис. 4.25 показана нормалізована пропускна здатність множини потоків від періоду атаки T (фактично фази обслуговування). Ситуація спостерігається аналогічна як і при одиночному потоці. Однак потрібно звернути увагу, що при T=1/*min*RTO пропускна здатність не дорівнює нулю (як було у випадку з одним потоком), і деякі потоки все ж таки змогли «прорватися» в модуль ІКС. Також зазначимо, що при частоті 2/*min*RTO пропускна здатність практично зводиться до нуля.

На рис. 4.26 показана нормалізована пропускна здатність для декількох потоків (від 1 до 3). Крива, позначена як «Атака відсутня», показує пропускну здатність кожного потоку без атаки. Крива, позначена як «Імітація атаки в MATLAB7 + Simulink», показує пропускну здатність кожного потоку в сумі з атакуючим імпульсом швидкістю 5 Мб/с, тривалістю 80–100 мс і періодом 0,5–1,0 с.

Рис. 4.25. Нормалізована пропускна здатність множини потоків від періоду атаки T

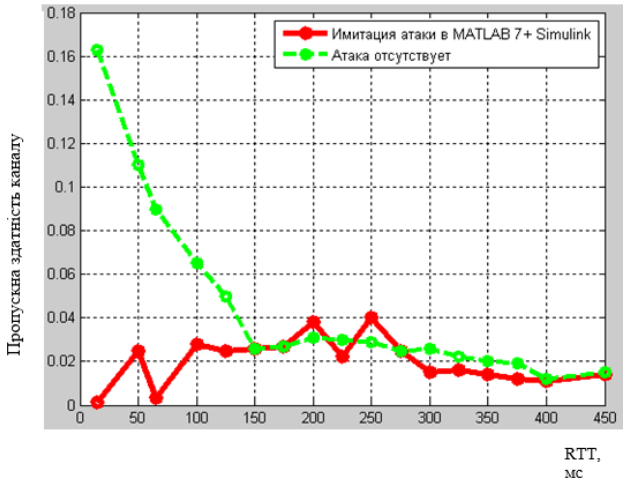


Рис. 4.26. Нормалізована пропускна здатність множини потоків

На рис. 4.27 показана залежність нормалізованої пропускної здатності від нормалізованої швидкості атаки. Як бачимо з графіка, навіть при низьких швидкостях спостерігається фільтрація потоків з малим RTT. Наприклад, якщо потік атакуючого займає одну третину всієї ємності каналу, то вже спостерігається істотне зниження смуги пропускання для потоків з малим RTT.

В даному випадку потоки з великим RTT відіграють роль фонового трафіку, допомагаючи атакуючому знизити швидкість атаки, тобто перевантажуючи додатково канал. Вищеописаний ефект дозволяє зробити висновок, що навіть низькошвидкісні потоки можуть справляти негативний ефект на трафік. Так, короткі імпульси на великій швидкості здатні зовсім вивести з ладу канал проходу трафіку, що збігається з результатами моделювання, викладеними в [4-15].

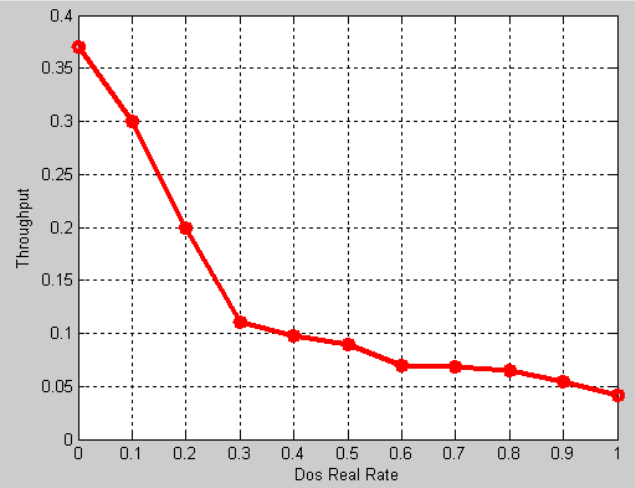


Рис. 4.27. Залежність ефективності атаки від її швидкості

У більшості випадків після захоплення управління атакуючому необхідно отримувати інформацію з хоста (наприклад, файли паролів для PLC, GSM-R і т. д.) і надсилати команди на хост. Для забезпечення прихованої передачі даних досить часто використовуються приховані канали. Основна ідея прихованих каналів полягає в тому, щоб передавати інформацію в невикористовуваних полях мережевих протоколів або змінювати некритично інформацію в мережевому протоколі. На сьогодні розроблена досить велика кількість

програм для створення прихованих каналів [4-15та ін.]. Оскільки завдання детального проектування ЗЗІ в межах цих досліджень не ставилося, нижче зупинимося тільки на найпростішій схемі моделювання блокування запиту в мережі підприємства при виявленні атаки типу «Відмова в обслуговуванні» при використанні прихованого каналу. Мережеве виявлення наявності прихованого каналу є досить складним [4-15]. При виявленні необхідно побудувати розв'язувальне правило (див. розділ 2), яке дозволить розділяти Intial Sequence Number (ISN), згенеровані оригінальним стеком, і ISN, згенеровані атакуючим.

Побудова розв'язувального правила можлива при врахуванні таких факторів. У більшості ОС ядро генерує ISN не випадковим чином. ISN є складною функцією від поточного часу і попереднього значення ISN. Атакуючий генерує ISN випадковим чином, оскільки шифрує дані на випадковому сеансовому ключі.

Розв'язувальне правило gov для розпізнавання атаки із використанням ДПРЗ ІБ будується так:

1) збираються ISN, згенеровані гарантовано чистим стеком;

2) зібрані ISN розглядаються як навчальна вибірка;

3) будується навчальна таблиця $P_i = ISN_i, T_i = ISN_{i+1}, i = N-1.$ (де N – число зібраних чистих ISN);

4) при перевищенні порогового значення ISN система приймає рішення, що тестовані пакети не відповідають моделі стека і обслуговування запитів переривається.

На рис. 4.28. показана схема підсистеми блокування запитів від АРМ ІКС та АСК при виявленні аномальної черги заявок, що надходять із терміналу.

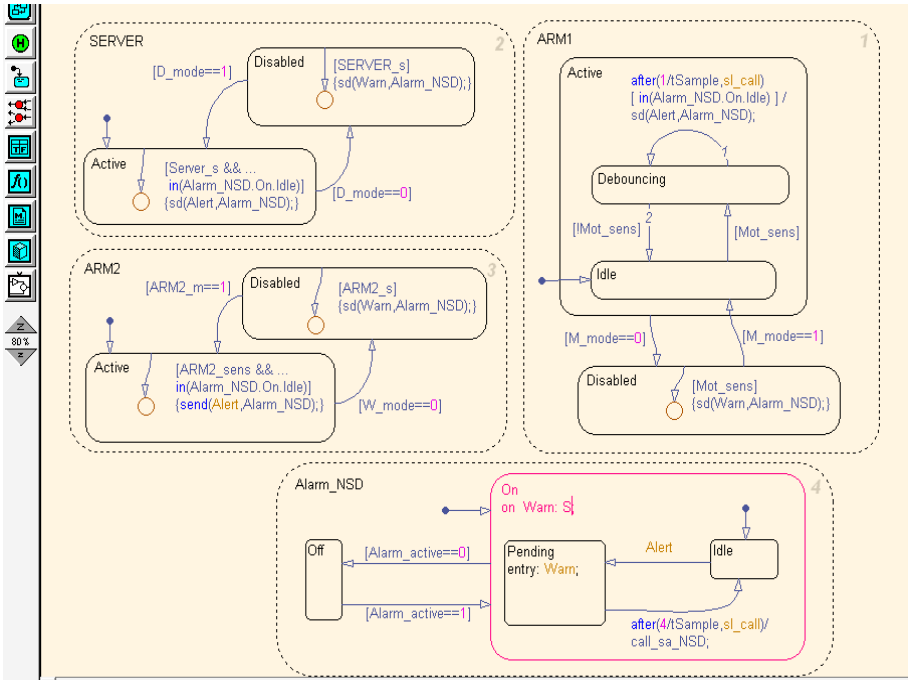


Рис. 4.28. Підсистема блокування запиту у від АРМ ІКС та АСК

В ході імітаційного моделювання досліджувалися режими роботи ІКС та АСК із ЗЗІ (блокування заявок при відхиленні запитів від «нормального» режиму) (розділ 3). Моделювався стаціонарний режим функціонування процесу генерації заявок на обслуговування. Заявки, що знаходяться в системі, систему не залишають, а продовжують обслуговуватися (заявки передаються в блок ДПРЗ ІБ) або очікують на обслуговування (заявки в черзі). Якщо потік блокований у модулі ДПРЗ ІБ, то за час $\Delta\tau$ з ймовірністю p_β потік розблокується, і заявки, які будуть згенеровані після цього моменту, знову надійдуть у систему на обслуговування.

У таблиці 4.1 наведені результати імітаційного моделювання заданої мережі (див. рис. 4.13, 4.15) в умовах атаки DoS/DDoS на сервер, АРМ та МАРМ. Середня помилка розрахованої оцінки ймовірності втрат заявок V_{zap} в результаті атаки не перевищує середньоквадратичне відхилення частоти втрат F_{zap} в серії експериментів.

Таблиця 4.1

Результати імітаційного моделювання заданої мережі

Кількість сеансів моделювання	n	10
Середнє значення частоти втрат заявок	V_{zap}	8,6E-2
Середньоквадратичне відхилення частоти втрат	F_{zap}	1,01E-3
Розрахована оцінка ймовірності втрат заявок	P_{zap}	8,41E-2
Середня помилка	ΔP_{zap}	3,9E-4

На рис. 4.29 показані результати моделювання ймовірностей інтелектуального розпізнавання загроз нападу на інформацію ($q01$) та блокування потоку заявок ($q10$) для випадку, коли заявки, що генеруються напівмарковським процесом, в систему не потрапляють, а блокуються при відхиленні запитів від стандартних параметрів.

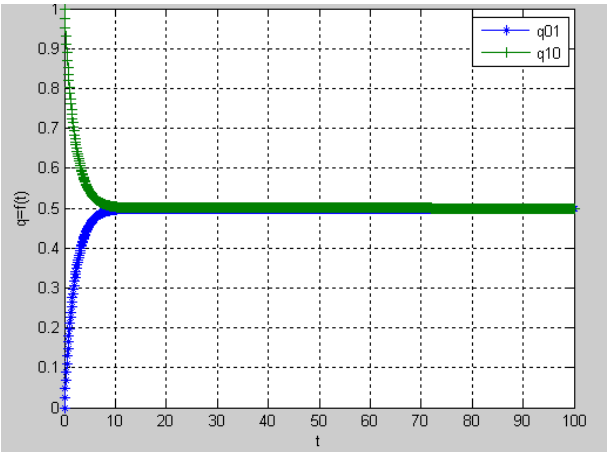
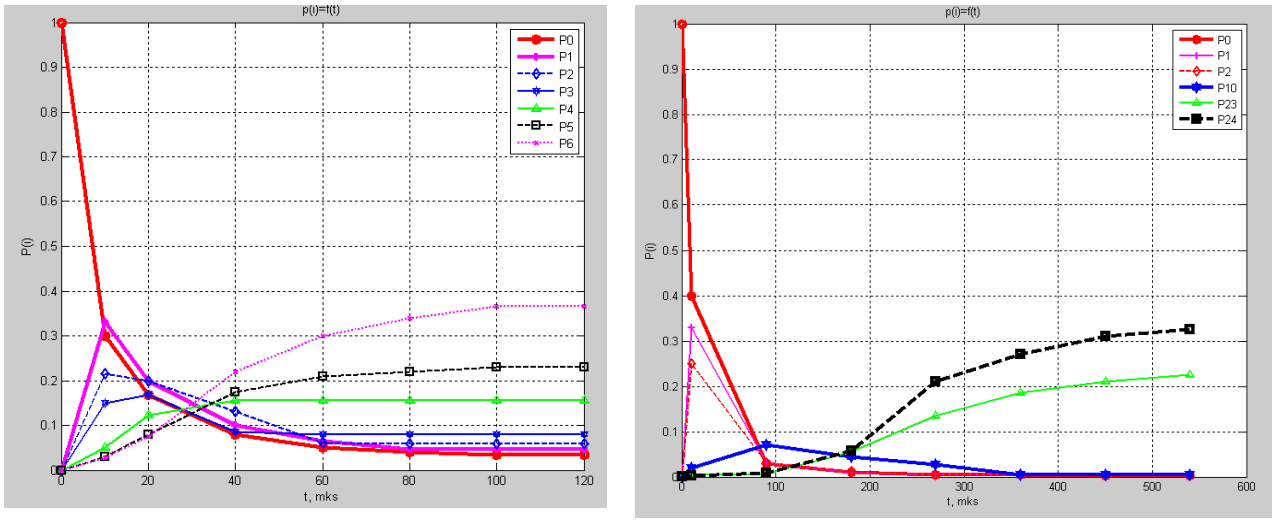


Рис. 4.29. Ймовірності розпізнавання загроз та блокування потоку заявок у ІКС

Оскільки всі вузли ІКС пов'язані комутаторами, зупинимося на деяких аспектах моделювання ймовірних їх станів при реалізації різних типів атак на ресурси ІКС.

На рис. 4.30 показані результати моделювання ймовірностей станів прийомного тракту комутатора ІС для випадку, коли інтенсивність вхідного

потоку перевищує інтенсивність передачі кадрів (атака DoS) (розмір буферної пам'яті LP = 4 кадри, рис. 4.30 а, LP = 22 кадри, 4.30 б).



а) б)

а) розмір буферної пам'яті LP = 4 кадри; б) розмір буферної пам'яті LP = 22 кадри

Рис. 4.30. Ймовірності станів передавального тракту комутатора ІКС при атаці DoS/DDOS

Як бачимо з відповідних графіків, найбільш ймовірною є успішна реалізація атаки DoS, спрямована на передавальний тракт портів комутатора. Однак час, необхідний для реалізації такої атаки, може бути суттєво збільшений (до 400 мкс), а сама ймовірність успішної реалізації атаки знижена за рахунок збільшення обсягу буферної пам'яті передавального тракту порту комутатора. Ці висновки підтверджуються експериментальними даними, отриманими в роботі [7].

На рис. 4.31, 4.32 представлені основні підсистеми імітаційної моделі ІКС, складені на підставі принципових структурних схем управління окремими видами, наприклад, транспорту та моніторингу ситуації, а також з урахуванням наявності в ІКС комп'ютерних компонентів – серверів ОЦ і БД, клієнтських станцій, телекомунікаційного обладнання і т.п. У зв’язку з великою кількістю підсистем, а також тієї обставини, що деякі підсистеми, наприклад, типова

ЛОМ, з параметрами, які налаштовуються (пропускна здатність, кількість клієнтських машин та ін.), містяться в бібліотеках MATLAB 7/2009, ми наводимо тільки автентичні схеми, складені із використанням компонентів Simulink.

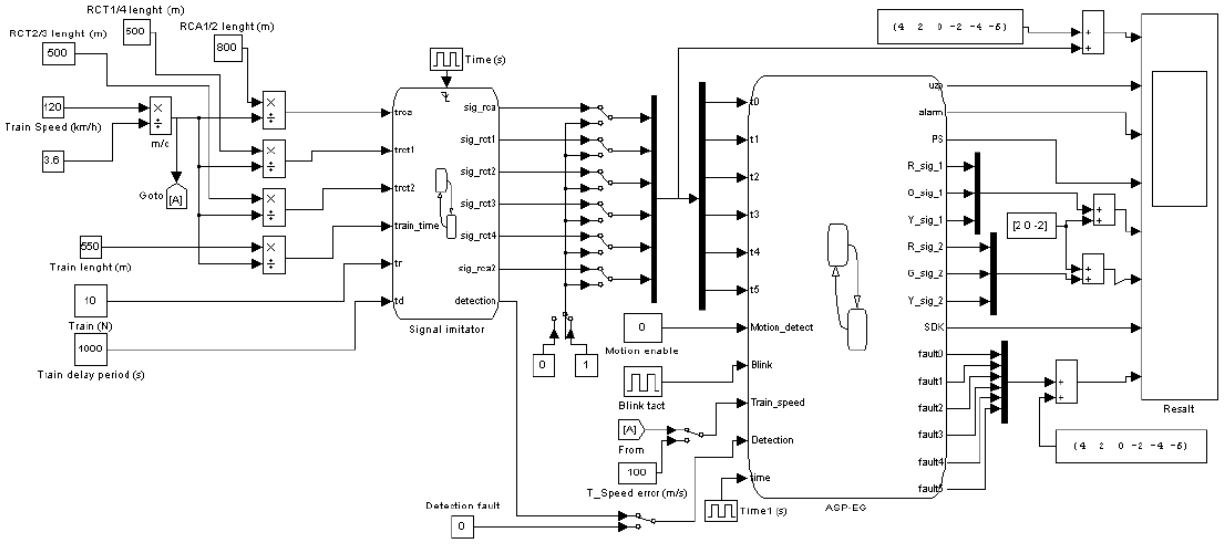


Рис. 4.31. Модель мережевої системи управління сигналізацією для здійснення руху ТЗ

На рис. 4.32 показані результати моделювання ймовірностей інтелектуального розпізнавання загроз KHI DoS ($q01, q10$) до мережевої системи управління сигналізацією для здійснення руху ТЗ та блокування потоку даних ($q11, q00$) системи для випадку, коли заявки, що генеруються напівмарковським процесом, в систему не потрапляють, а блокуються при відхиленні запитів від стандартних параметрів.

У таблиці 4.2 наведені результати імітаційного моделювання мережевої системи управління сигналізацією для здійснення руху ТЗ в умовах атаки DoS/DDoS на АКС.

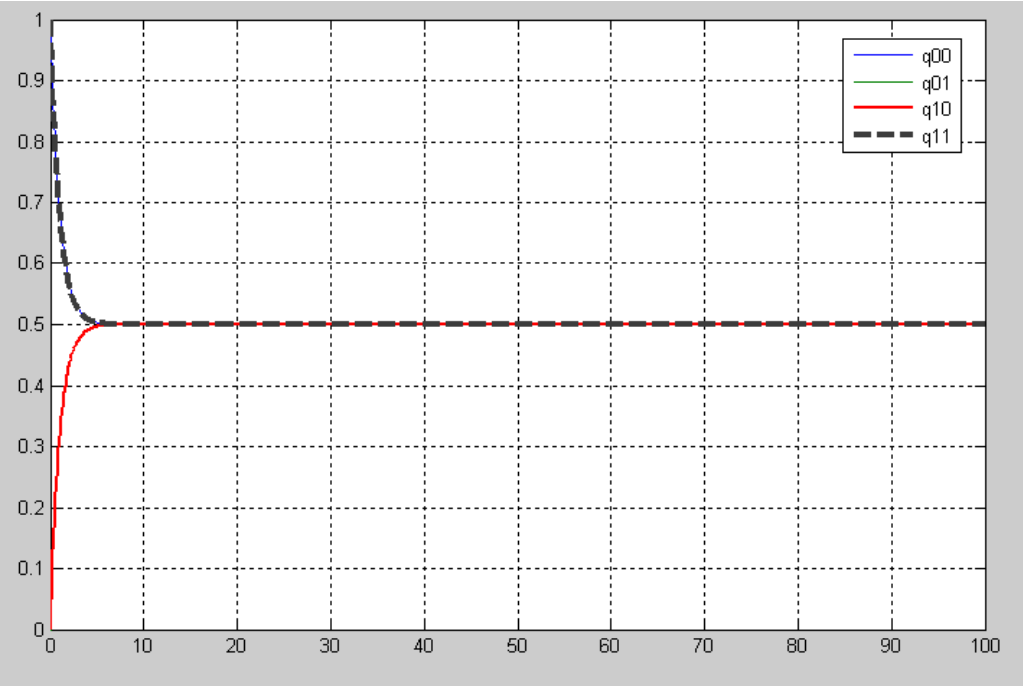


Рис. 4.32. Ймовірності розпізнавання загрози КНІ DoS до мережевої системи управління сигналізацією для здійснення руху ТЗ та блокування потоку даних системи

Таблиця 4.2

Результати імітаційного мережевої системи управління сигналізацією для здійснення руху ТЗ в умовах атаки DoS/DDoS

Кількість сеансів моделювання	n	15
Середнє значення частоти втрат заявок	V_{zap}	4,6E-2
Середньоквадратичне відхилення частоти втрат	F_{zap}	1,02E-3
Розрахована оцінка ймовірності втрат заявок	P_{zap}	4,41E-2
Середня помилка	ΔP_{zap}	3,87E-4

Оскільки одними з найбільш уразливих каналів відпливу інформації в ІКС та АСК є джерела відеоспостереження і моніторингу, а також відеосервер, зупинимося на деяких аспектах моделювання їх роботи більш детально. При мережевому типі систем відеоспостереження уразливості виходять вже на

рівень високих технологій, особливо якщо система відеоспостереження інтегрована в комп'ютерну мережу компанії або працює по базі бездротового або широкосмугового зв'язку. Тому доцільно дослідити всі ці нові уразливі місця в структурі ІКС. Очевидно, найбільш небезпечною буде ситуація, коли зловмисник, отримавши доступ до відеосервера, зможе потрапити в ЛОМ компанії, тому що через відеосервер можна потрапити в мережу, оминаючи звичайні брандмауери, (прецедент мав місто, коли, наприклад, були відключені камери спостереження та видалена інформація з БД комплексу фіксації швидкісного режиму – «Стрілка-М»). На рис. 4.33, 4.34, наприклад, показані схеми підсистеми виведення відеоінформації на монітор системи спостереження та управління рухом залізничного, автомобільного та морського транспорту, а також підсистема передачі відеозображення по бездротовим або дротовим каналам на сервери ІКС.

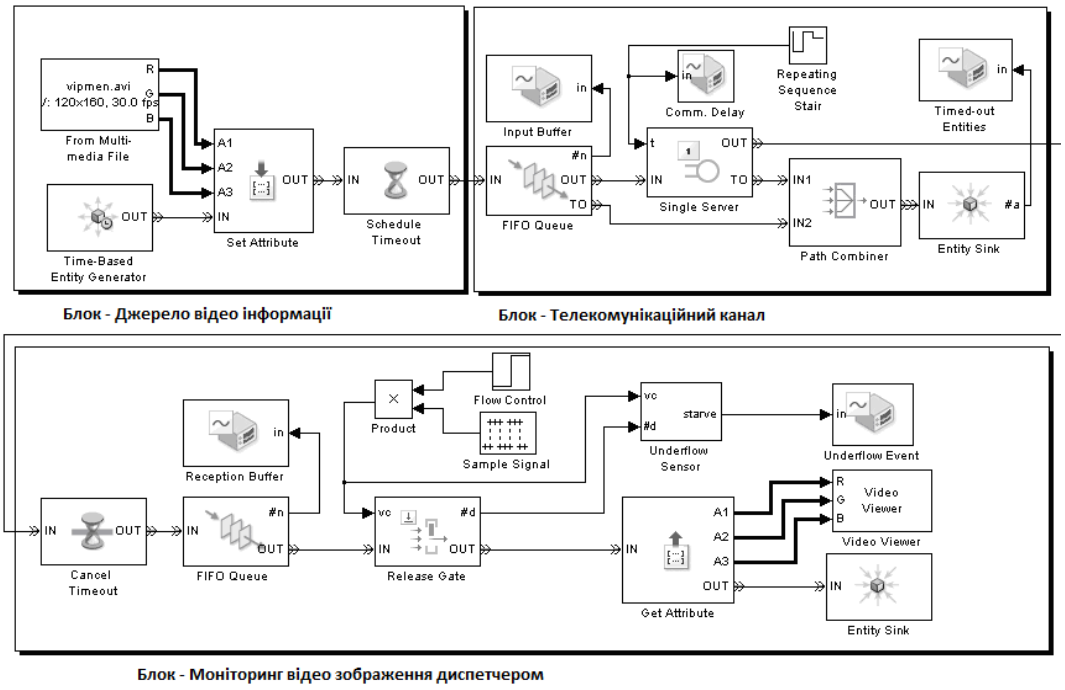
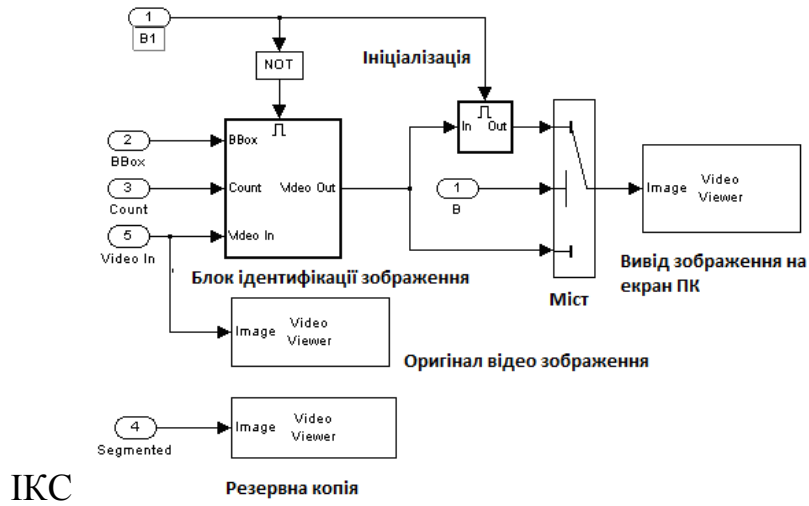
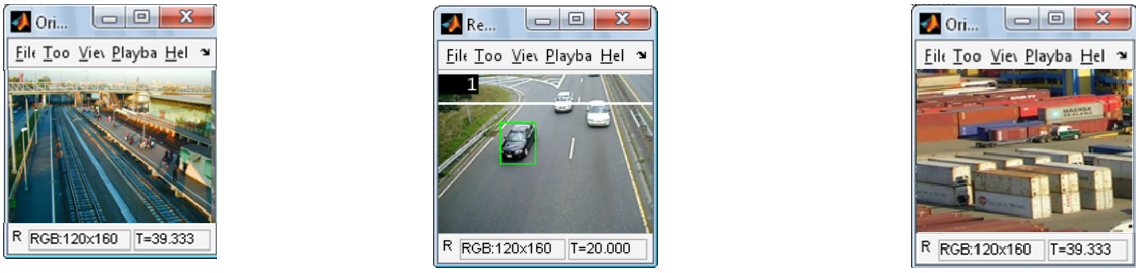


Рис. 4.33. Модель мережевої системи відеоспостереження і моніторингу для



ІКС

а)



б)

Рис. 4.34. Підсистема виведення відеоінформації с камер спостереження

У таблиці 4.3 наведені результати імітаційного моделювання мережевої системи відеоспостереження і моніторингу для ІКС в умовах розпізнавання КНІ DoS/DDos.

Таблиця 4.3

Результати імітаційного моделювання мережевої системи відеоспостереження і моніторингу в умовах розпізнавання КНІ DoS/DDos

Кількість сеансів моделювання	n	15
Середнє значення частоти втрат заявок	V_{zap}	6,34E-2
Середньоквадратичне відхилення частоти втрат	F_{zap}	1,17E-3
Розрахована оцінка ймовірності втрат заявок	P_{zap}	6,72E-2
Середня помилка	ΔP_{zap}	1,34E-3

За результатами спостережень можна відзначити, що середня помилка розрахованої оцінки ймовірності втрат заявок V_{zap} не перевищує середньоквадратичне відхилення частоти втрат F_{zap} в серії експериментів.

На рис. 4.35 і 4.36 показані результати моделювання ймовірностей переходів станів сервера ІКС та АСК для різних варіантів вхідних потоків і станів міжмережевого екрану (див. рис. 3.3).

Тут потрібно врахувати, що інтенсивності переходів мають «стандартне значення» (див. розділ 3), тобто в ході моделювання не враховувалися варіанти надходження пакетів при варійованій ширині смуги пропускання або швидкості пакета (потоку).

Отримавши ймовірності станів за кожним із вузлів системи, на наступному етапі ми досліджували залежність функції розподілу часу до виходу системи з «нормального» режиму функціонування при різних типах потоків даних (що мають різну інтенсивність і пріоритет виконання), в тому числі неоднорідних, і часу для проведення зловмисником нападу на інформацію.

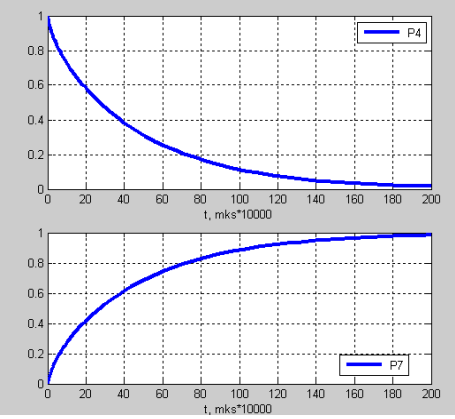


Рис. 4.35. Ймовірності переходів сервера і МЕ в стани S_4 і S_7

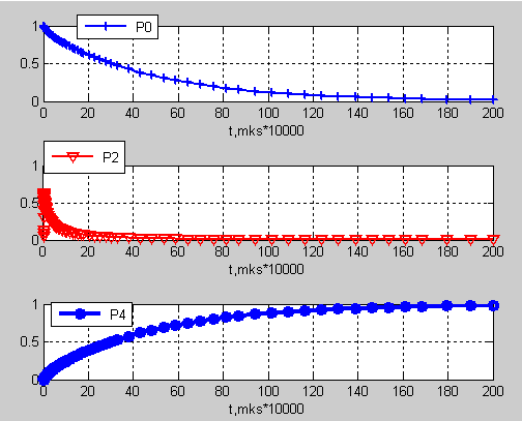


Рис. 4.36. Ймовірності переходів сервера і МЕ в стани S_{ME0}, S_2 і S_4

Правила для ДПРЗ змінювалися у відповідному блоці Fuzzy Logic Toolbox, див. рис. 4.37. Основні результати моделювання показані на графіках 4.38.

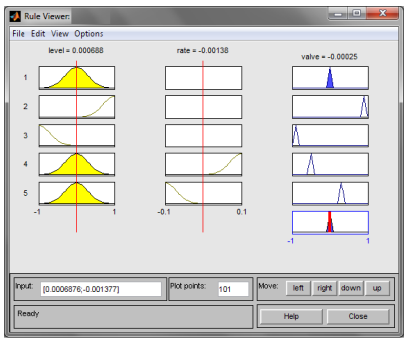
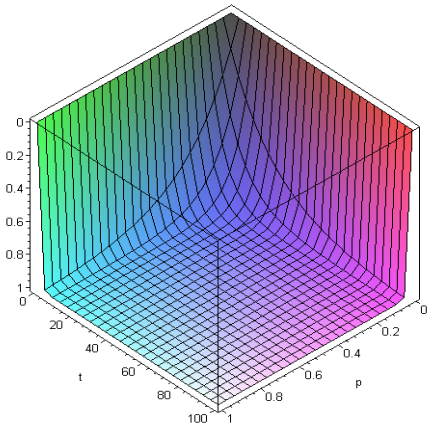
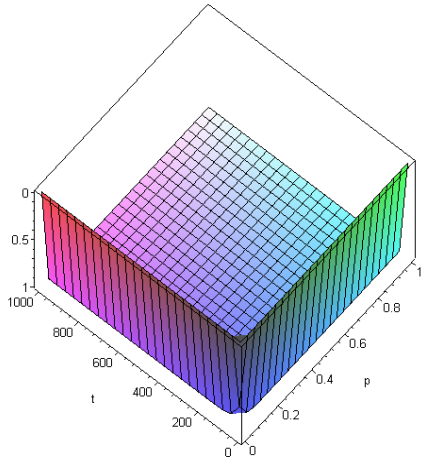


Рис. 4.37. Графічний блок зміни правил для ДПРЗ



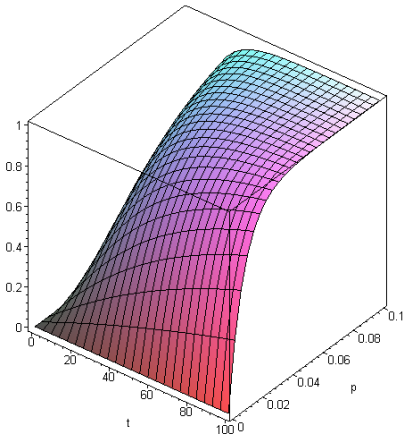
t=0..100, P(S_1)=0..1

а)



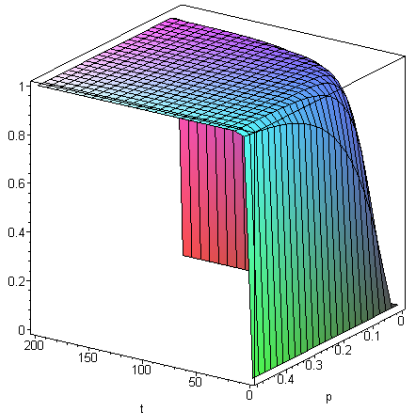
t=0..1000, P(S_1)=0..1

б)



t=0..100, P(S_1)=0..0.1

в)



t=0..200, P(S_1)=0..0.5

г)

Рис. 4.38. Залежність функції розподілу часу до виходу системи з «нормального» режиму функціонування при різних типах потоків даних і часу для проведення зловмисником КНІ

На рис. 4.39 представлений графік залежності теоретичної оцінки ймовірності втрати заявки в ІС від кількості кроків запропонованої ітераційної процедури (3.17–3.22). Представлений графік дозволяє зробити висновок про необхідну кількість ітерацій для забезпечення заданої точності імітаційної моделі.

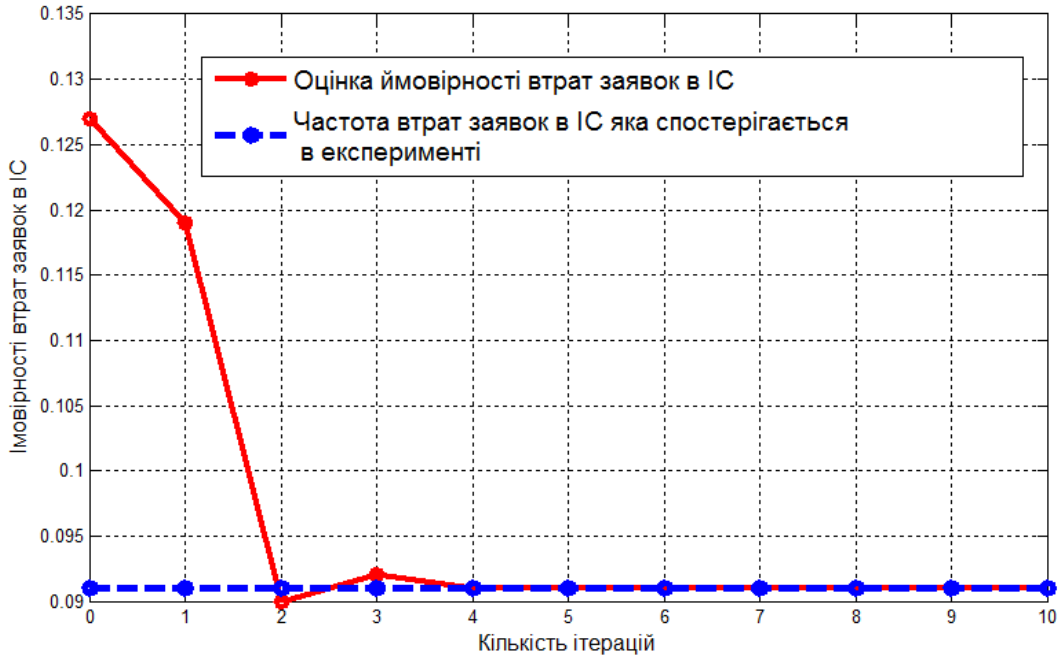


Рис. 4.39. Залежність теоретичної оцінки ймовірності втрати заявки від кількості кроків ітераційних процедур (вирази 3.16 – 3.22)

Розроблена імітаційна модель дозволяє отримувати адекватну оцінку частоти втрати заявок в мережі у випадку, якщо ІС працює в стаціонарному режимі (закон розподілу Пуасона). Під час виникнення КНІ DoS/DDoS один або декілька вузлів ІС виходять з стаціонарного режиму на деякий час, після чого встановлюється стаціонарний режим з іншими параметрами. Таким чином, оцінка часу переходу між режимами ІС має визначальне значення. Очевидно, що час переходу сильно залежить від топології мережі і параметрів вузлів ІКС.

Як показує аналіз отриманих залежностей, а також даних, наведених у 3-му розділі роботи, при використанні зловмисником тактики присвоювання малоінтенсивному потоку заявок високого пріоритету і достатньому часі на

проведення КНІ, можна збільшити ймовірність проникнення в систему і переміщення по станам $S_1 - S_7$ (див. рис. 3.3), не змінюючи при цьому параметрів потоку, що має в системі найбільшу інтенсивність і високий пріоритет.

Таким чином, для проведення успішної атаки на інформаційні ресурси ІКС, зокрема, типу «відмова в обслуговуванні», не обов'язково створювати велику кількість запитів до серверів, систем GSM-R, VSAT, PLC або знижувати смугу пропускання трафіку. Можна з досить високим ступенем ймовірності успіху експлуатувати уразливості, пов'язані зі створенням малоінтенсивного пріоритетного потоку, наприклад, варіюючи такими параметрами, як швидкість пакета (низькошвидкісні DoS/DDoS атаки); кількість пакетів з нульовою частотою щодо RTT; тривалість імпульсу та ін.

За допомогою імітаційної моделі (ІМ) проведено перевірку вірогідності результатів реалізації КНІ DoS/DDoS у інфокомунікаційній мережі, параметри елементів якої розраховувалися за допомогою запропонованої моделі інтелектуального розпізнавання загрози реалізації атаки каналами GSM-R, VSAT або звичайних кабельних або бездротових мереж ІКС. У якості вихідних даних використовувалися результати вимірювань параметрів отриманих реалізацій вхідних потоків у ІМ. Експеримент з імітаційного моделювання та розрахунок пропускної здатності каналів (наприклад, GSM-R або VSAT), які використовуються у ІКС, проводився для різних наборів реалізацій неоднорідних потоків k_1 та k_3. Результати аналізу часу затримки або втрати заявки в мережі, отриманого за допомогою імітаційного моделювання, та порівняння його з часом затримки, що очікувався, наведено у таблиці 4.4. Відповідно, $T_{\text{ср.пр.}}$ та $P_{\text{пр.}}$ – припустимі значення затримки та ймовірності втрат відповідно, $T_{\text{ср.к.}}$ та $P_{\text{к.}}$ – відповідні параметри мережі, розрахованої класичним методом, $T_{\text{ср.з.}}$ та $P_{\text{з.}}$ – відповідні параметри мережі, розрахованої за допомогою запропонованих моделей. З аналізу результатів робиться висновок про адекватність зроблених розрахунків параметрів елементів мережі ІС об’єкту інформаційної безпеки.

Таблиця 4.4

Значення ймовірносно - часових характеристик при КНІ DoS/DDoS

Номер реалізації	$T_{ср.пр.}$, мс	$T_{ср.к.}$, мс	$T_{ср.з.}$, мс	$P_{пр.}$	$P_{к.}$	$P_{з.}$
1	10	11,7	10,2	$5x10^{-8}$	$6,28x10^{-8}$	$4,93x10^{-8}$
2	20	24,7	20,3	$7x10^{-8}$	$7,79x10^{-8}$	$7,12x10^{-8}$
3	50	61	49,5	$3x10^{-6}$	$3,6x10^{-6}$	$3,15x10^{-6}$
4	100	97,3	98,4	$1,5x10^{-6}$	$2,36x10^{-6}$	$3,07x10^{-6}$
5	150	107,3	101,4	$1,7x10^{-6}$	$2,56x10^{-6}$	$2,98x10^{-6}$
6	200	111,3	119,4	$1,8x10^{-6}$	$2,7x10^{-6}$	$2,81x10^{-6}$
7	250	125,3	128,4	$1,95x10^{-6}$	$2,8x10^{-6}$	$2,47x10^{-6}$
8	300	147,3	148,4	$2x10^{-6}$	$2,9x10^{-6}$	$2,03x10^{-6}$
9	350	180,3	155,4	$9,1x10^{-5}$	$9,05x10^{-5}$	$9,29x10^{-5}$
10	400	247,3	190,4	$9,7x10^{-5}$	$9,9x10^{-5}$	$9,87x10^{-5}$

Аналіз результатів проведеного експерименту, наведених у таблиці 4.4, дозволяють зробити висновок, що запропонована модель ДПРЗ у випадку використовування неоднорідних потоків запитів має більшу точність, ніж існуючи моделі, приблизно на 5–7 %.

Для проведення аналізу отриманих в ході імітаційного моделювання результатів представимо отримані данні у вигляді таблиці, див. табл. 4.5.

Проаналізувавши отримані дані, можна зробити висновок, що всі імітаційні моделі компонентів ІКС та відповідні ДПРЗ, надали досить точні результати. Незважаючи на різницю їх структурної реалізацій, імітаційні моделі впоралися з поставленою задачею виявлення аномалій та інтелектуального розпізнавання загроз при нападах на ІКС.

Таблиця 4.5

Зведені результати імітаційного моделювання

№	Імітаційна модель	Вихідні дані					
		Аномальний стан/Помилки першого роду/Помилки другого роду			Відсутність атаки/Помилки першого роду		
		Дерево рішень	Алгоритм найближчого кластера	ДПРЗ+ нечіткі бази знань	Дерево рішень	Алгоритм найближчого кластера	ДПРЗ+ нечіткі бази знань
1	Сегмент комп'ютерної мережі підприємства	0,74/ 0,125/ 0,035	0,85/ 0,114/ 0,05	0,92/ 0,102/ 0,029	0,35/ 0,13	0,56/ 0,15	0,86/ 0,09
2	Мережева система управління сигналізацією для здійснення руху ТЗ	0,62/ 0,14/ 0,07	0,655/ 0,12/ 0,07	0,872/ 0,112/ 0,035	0,78/ 0,13	0,89/ 0,16	0,91/ 0,12
3	Мережева система відео-спостереження і моніторингу	0,69/ 0,132/ 0,062	0,81/ 0,15/ 0,05	0,905/ 0,12/ 0,027	0,42/ 0,15	0,77/ 0,11	0,94/ 0,095

4.3. Висновки до розділу 4

В результаті проведених в даному розділі навчального посібника можна зробити наступні висновки.

1. Визначено, що використання імітаційного моделювання, яке об'єднує між собою різнорідні математичні моделі елементів, що входять до складу ІКС,

є одним із методів, які дозволяють оцінити ЗЗІ ІКС та АСК та її реакцію на спроби НСД (збурення) за рядом показників.

2. Встановлено, що за допомогою імітаційного моделювання в середовищі MATLAB та Simulink при створенні ЗЗІ ІКС можуть вирішуватися завдання з визначення шляхів удосконалення захисту інформації та ІБ ІКС, АСК та ін. на підставі аналізу різних варіантів технічної, технологічної, а також організаційної перебудови та дослідження наслідків прийнятих рішень.

3. Проаналізовано можливість написання і підключення власних модулів в середовищі MATLAB та Simulink для реалізації математичних моделей, що описують ймовірнісні стани системи та її ентропію.

4. Показана можливість зміни параметрів імітаційних моделей під час проведення експериментів в MATLAB та Simulink з моделювання процесів нападу на компоненти ІКС.

5. Встановлено, що програмна реалізація математичних моделей ЗЗІ ІКС, дозволяє отримати наочні уявлення процесів нападу на ІС та АСК. Визначено важливі характеристики СМО (час простою лінії, середній час передачі даних по лінії від кожної станції і т. п.).

6. Виконано імітаційне моделювання КНІ на сервери ІКС, мережеву систему відеоспостереження і моніторингу, мережеву систему управління сигналізацією для здійснення руху ТЗ.

7. Перевірено роботу алгоритму вирішального правила для підсистеми блокування запитів при виявленні аномальної черги заявок, що надходять з терміналів АРМ, МАРМ.

8. Встановлено, що при використанні зловмисником тактики присвоювання малоінтенсивному потоку заявок високого пріоритету і достатньому часі проведення нападу на інформацію, можна збільшити ймовірність проникнення в систему, не змінюючи при цьому параметри потоку, що має в системі найбільшу інтенсивність і високий пріоритет.

9. Встановлено, що всі імітаційні моделі компонентів ІКС та відповідні ДПРЗ, надали досить точні результати. Незважаючи на різницю їх структурної

реалізацій, імітаційні моделі впоралися з поставленою задачею виявлення аномалій та інтелектуального розпізнавання загроз при нападах на ІКС.

10. Встановлено доцільність об'єднання в одному територіальному центрі аналізу всіх даних про стан ІКС та АСК для спільного проведення високоефективної політики у сфері ІБ для зацікавлених суб'єктів господарювання.

11. Для подальшого вирішення проблеми захисту інформації та підвищення ІБ ІКС, потрібно зосередитися на розгляді завдань оптимізації складу ЗЗІ та структурно-технологічного резерву критично важливого програмного та інформаційного забезпечення ІКС та АСК, а також удосконалити економетричну модель підтримки прийняття рішення по вибору оптимальної стратегії управління інвестиційним проектуванням у систему інформаційної безпеки та захисту інформації господарюючого суб'єкта.

РОЗДІЛ 5
ОПТИМІЗАЦІЯ СКЛАДУ КОМПЛЕКСІВ ЗАСОБІВ ЗАХИСТУ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНОГО СЕРЕДОВИЩА

Як показує досвід експлуатації ІС у різних галузях, гарантоване стовідсоткове забезпечення безпеки – справа дорога й не завжди доцільна, оскільки:

1) навіть зроблена на сьогодні СЗІ не зможе протидіяти загрозам, які можуть виникнути надалі [4-15 та ін.];

2) вартість комплексного захисту може виявитися значно вищою, ніж вартість інформаційних ресурсів, що захищаються.

При розробці сучасних СЗІ доводиться враховувати велику кількість факторів, що впливають на ефективність їх функціонування, а це ускладнює пошук аналітичних оцінок з вибору узагальненого критерію оптимальності їх структури. Тому в даному розділі роботи ми зупинимося на питаннях оптимізації складу обраної структури СЗІ ІКС за різними критеріями. З методології системного підходу відомо, що математичний опис складної системи, до якої належить і СЗІ, здійснюється шляхом ієрархічної розбивки її на елементарні складові. При цьому в математичні моделі вищих рівнів як частинні критерії завжди повинні включатися узагальнені критерії нижчестоящих рівнів. Отже, те саме поняття стосовно нижчого рівня може виступати як узагальнений критерій (мети), а стосовно вищого – як частинний критерій (завдання).

У цілому, комплекс СЗІ ІКС повинен відповідати таким вимогам:

1) забезпечення захисту від усіх потенційних загроз інформації;

2) застосування всього спектра засобів захисту інформації;

3) захист повинен проводитися на всіх етапах життєвого циклу об'єкта інформатизації.

5.1. Оптимізації структурно-технологічного резерву програмного забезпечення інформаційно-комунікаційного середовища

Однією з основних умов ефективного функціонування розподілених комп'ютерних систем, зокрема ІКС, є забезпечення необхідного рівня схоронності їх функціональних та інформаційних ресурсів. Різке збільшення обсягів опрацьованих даних і необхідність їх спільного використання більшим числом споживачів призводять до необхідності вводити до складу комп'ютерних систем підсистеми розподіленого зберігання даних, які найчастіше будуються на обладнанні мереж зберігання даних (МЗД). Застосування методів розподіленого зберігання даних в ІКС або АСК, обумовлює виникнення ряду нових завдань забезпечення безпеки інформаційних масивів (ІМ) та ПЗ, але водночас і розширює можливості створення нових засобів і механізмів захисту. Зокрема, засоби розподіленого зберігання даних можуть бути ефективно застосовані для підвищення показників доступності й цілісності інформаційних ресурсів ІКС.

Наприклад, тільки у складі АСК ВП УЗ-Є та АСК ПП УЗ (див. розділ 1) задіяно більше 60 тис. програмних компонентів (програм, таблиць, та ін.). Щодоби в системах АСК ВП УЗ-Є та АСК ПП УЗ опрацьовується понад 800 тис. запитів. На сьогодні системи системах АСК ВП УЗ-Є та АСК ПП УЗ обслуговують понад 30 тис. користувачів [1].

Для забезпечення надійності ПЗ та ІМ ІКС або АСК, а, отже, і її безпеки, необхідно планувати їх регулярне резервне копіювання. Резервне копіювання потрібно виконувати після створення бази даних, після того як вона була завантажена даними, після створення індексів або коли виконані певні оператори, які не журналізуються.

Аналіз типів резервного копіювання в системах керування базами даних SQL Server 2000/2012, InterBase 7/XE і Oracle 8/11 [4-15] дозволяє виділити три основні рівні резервного копіювання, які використовуються в комбінації з

різними схемами ротації носіїв: повне резервування й два типи часткового резервування (диференціальне й додаткове).

Найбільш ефективним підходом до вирішення проблеми забезпечення необхідного рівня схоронності інформації в умовах постійно зростаючого переліку загроз, засобів і методів їх реалізації є резервування.

Класичні підходи до забезпечення високої схоронності інформаційних масивів засновані на методах ідентичної або неідентичної надмірності (відповідно зберігання копій або зберігання передісторій), згідно з якими, зазвичай, для конкретної системи обирається ефективна стратегія резервування [14]. Безумовною перевагою більшості класичних методів резервування є їхня простота, недоліками – високий ступінь дублювання інформації, неефективне використання ємності магнітних носіїв, у ряді випадків – неприпустимо великий час на відновлення ІКС або АСК у разі нападу на інформацію.

Системи розподіленого зберігання даних відкривають можливості для широкого використання методів відновного резервування, що поєднують (порівняно невисоку) надмірність із уведенням аналітичних залежностей між блоками даних, що представляють інформацію, яка зберігається.

У зв'язку з тим, що можливі топології мереж ІКС або АСК можуть бути не рівноймовірними, у даному розділі ми наводимо математичну модель оптимізації структурно-технологічного резерву за критерієм мінімуму ймовірності невирішення завдання.

Тобто, відповідно до постановки завдання, необхідно знайти такі значення $x_{nm}^{um^*}$, які є

$$\min_{\left\{ X_{u\,num}^{um^*} \right\}} \prod_{un=1}^{N} \prod_{um=1}^{M_{inf}} \left[\prod_{um^*}^{M_{inf}} \varphi_{unum}^{um^*} \prod_{un'=1}^{N_{po}} \prod_{um'=1}^{M_{inf}} \prod_{um_1^*=1}^{M_{inf}{}^*} \prod_{um_2^*=1}^{M_{inf}{}^*} P_{um_1^*\,um_2^*}^{unumun'\,um'} \right], \quad (5.1)$$

де um^* – вузол у системі, наприклад, у ЛОМ ІКС або АСК;

M_{inf} – кількість інформаційних масивів у ІКС або АСК;

N_{po} – кількість програмних модулів у ІКС або АСК,

при обмеженнях:

– на структурне дублювання модулів $X^{um_1^*}_{unum}\ X^{um_2^*}_{un'um'} = 0$ для $\forall un, um, un', um', um_1^*, um_2^*$, для яких виконуються умови $C_{um_1^* um_2^*} = 0$, $\varphi^{un'um'}_{unum} \neq 0$;

– на розподіл окремих модулів за окремими вузлами $X^{um^*}_{unum}=1$, для виділених $unum$ операційних модулів і um^* - *их* вузлів;

– на найбільший можливий час вирішення завдання, наприклад, перерахунок графіка руху ТС на маршруті [4-15]

$$\sum_{un=1}^{N_{po}} \max_{\left\{\substack{um^* \\ um}\right\}} \left[X^{um^*}_{unum}\, \theta_{unum}\, \lambda_{unum} \right] + \sum_{un=1}^{N_{po}} \sum_{un'}^{N_{po}} \sum_{um=1}^{M_{inf}} \sum_{um'=1}^{M_{inf}} \max_{\left\{\substack{um_1^* \\ um_2^*}\right\}} \left[X^{um_1^*}_{unum}\, X^{um_2^*}_{un'um'}\, \varphi^{un'um'}_{unum}\, \frac{1}{C_{um_1^* um_2^*}} \right] \le T^*, \tag{5.2}$$

де T^* – максимально можливий час вирішення завдання;

$\sum_{un=1}^{N_{po}} \sum_{um=1}^{M_{inf}} x^{um^*}_{umun}\, f_{unum} \le V_{um^*}, \qquad \forall um^*,\ um^* = \overline{1, M_{inf}}.$ – на максимальний обсяг зовнішньої пам'яті вузлів ІКС або АСК.

Поставлене завдання належить до задач нелінійного програмування з булевими змінними. Вона може бути зведена до цілочисельної оптимізаційної задачі лінійного програмування шляхом логарифмування цільової функції й спрощення обмеження (5.2) до лінійного [4-15 і ін.].

Використання структурно-технологічного резервування програмного й інформаційного забезпечення (ПЗ й ІЗ) завдань, вирішуваних ІКС або АСК, дасть змогу підвищити ІБ ІКС в умовах дії дестабілізуючих факторів з

урахуванням обмежень на максимальний час вирішення завдань. Реалізація запропонованої моделі синтезу структурно-технологічного резерву дозволить у перспективі для конкретних ЛОМ визначати норми резервування на структуру й обсяг резерву ПЗ та ІЗ розподілених завдань ІКС або АСК, що функціонують на основі ЛОМ, з урахуванням установленої нижньої межі безпеки. Також можуть бути визначені норми на резервний обсяг зовнішньої пам'яті у разі паралельного опрацювання.

При вирішенні завдання схоронності інформації методами резервування й відновлення даних пропонується використовувати віртуально-відновний резерв, до складу якого входять як самі дані, так і їх копії та/або передісторії.

Як наслідок, формалізується завдання визначення оптимального змісту віртуально-відновного резерву і його розміщення по вузлах ІКС або АСК (для критично важливих вузлів), вирішуване на етапі передпроектного аналізу (див. розділ 1).

У випадках, коли ІКС та АСК складаються з однорідних елементів за ознакою ступеня ризику виникнення надзвичайних ситуацій, як основний критерій синтезу віртуально-відновного резерву пропонуємо використовувати максимальний критерій рівномірного розподілу виграшу за вузлами ІКС або АСК, критерій мінімуму ступеня віртуальності резерву та ін.

Задача проектування віртуально-відновного резерву за першим критерієм має такий вигляд:

Знайти: $$max \min_{um} \sum_{un=1}^{N_{po}} C_{unum} \cdot x_{unum},$$

при обмеженнях:

– на ступінь віртуальності резерву для ПЗ і ІЗ в ІКС або АСК

$$1 - \prod_{un=1}^{N_{po}} (1 - \alpha_{un} \sum_{um=1}^{M_{inf}} \sum_{um=1}^{M_{inf}} x_{unum} \cdot \lambda_{unum} \cdot t_{umum'}^{un}) \le R_{vir},$$

де R_{vir} – максимально припустимий ступінь віртуальності резерву;

– на відносний час коригувань інформаційних елементів

$$\sum_{un=1}^{N_{po}} \sum_{um=1}^{M_{inf}} \sum_{um'=1}^{M_{inf}} x_{unum} \cdot \alpha_{un} \cdot \lambda_{unu\,m'}^{un} \le T_{кор\ (max)} \ ,$$

де $T_{кор(\max)}$ – максимально припустимий відносний час коригування інформаційного елемента;

– на обсяг зовнішньої пам'яті im^* - ого вузла ІКС або АСК

$$\sum_{un=1}^{N_{po}} x_{unum} \cdot b_{un} \le B_{um}^*, \quad \forall um^*, um^* = \overline{1, M_{inf}},$$

де B^*_{im} – максимально припустимий обсяг пам'яті im^* *-ого* вузла для зберігання інформації;

– на відсутність дублювання інформаційного елемента у вузлах ІКС або АСК

$$\sum_{un=1}^{N_{po}} x_{unum} = 1, \quad \forall um^*, um^* = \overline{1, M_{inf}}.$$

Результатом вирішення завдання синтезу віртуально-відновного резерву є оптимальний за заданими критеріями інформаційний склад масивів даних (ІМ), розміщених по вузлах ІКС або АСК. Використання віртуально-відновного резервування даних в ІКС, що функціонують на базі ЛОМ, засноване на більш гнучкому використанні поняття якості бізнес – інформації об’єкту інформаційної безпеки, дозволяє підвищити оперативність опрацювання даних, а також їх схоронність в умовах дії дестабілізуючих факторів, наприклад, при КНІ ІКС або АСК, або копіювання БД.

Одним з істотних факторів, що визначають стійкість інформаційно-обчислювального процесу й функціонування системи в цілому до дії дестабілізуючих факторів, є раціональне розміщення інформаційних ресурсів

ІКС або АСК, у тому числі з урахуванням стійкої тенденції до інтеграції окремих компонентів АС із мережами загального користування. Вирішення зазначеного завдання сприяє вибору оптимальних інженерних рішень на різних етапах проектування, експлуатації, удосконалювання й розвитку ІКС або АСК. Їхнє вирішення забезпечує як аналіз, так і оптимальний синтез системи обчислювальних засобів і їх компонентів.

Для скорочення розмірності завдань оптимізації інформаційно-обчислювального процесу ми розглядаємо ІКС або АСК як сукупність вкладених контурів керування. Основним призначенням цієї розбивки є така організація системи, яка призводить до необхідності внесення змін або в один з її елементів, або, у крайньому разі, в мінімальне їх число.

Кожному контуру керування повинна відповідати своя деталізація інформаційно-обчислювального процесу, що зростає з рухом униз (у напрямку більш докладного опису процесів) і дозволяє здійснювати взаємне ув'язування основних елементів інформаційно-обчислювального процесу на відповідному рівні. Число контурів керування повинне визначатися виходячи із практичних потреб проведеного дослідження. При такому підході вирішення будь-якого достатньо складного завдання може бути досягнуте в результаті послідовного уточнення значень параметрів системи і її структурних компонентів за допомогою розрахунків на сукупності математичних моделей.

На етапі проектування пропонується вирішення завдань оптимізації інформаційно-обчислювального процесу (розподіл програмних модулів (ПМ) і ІМ, а також їх резерву) здійснювати за принципом «зверху-униз». У контурах керування нижнього рівня (вузол ІКС або АСК) завдання оптимізації інформаційно-обчислювального процесу виражаються у визначенні складу й структури ІКС і розподілі завдань (програм), ІМ (баз даних) та їх відновного резерву між декількома ЕОМ з урахуванням їхнього пріоритету й інтенсивності вирішення, обмежень на обсяг пам'яті і час вирішення кожного завдання, а також у визначенні необхідного для забезпечення заданого рівня показника схоронності інформації обсягу відновного резерву кожного ПМ і ІМ. Таким

чином, вирішення послідовності завдань оптимізації дозволяє визначати й уточнювати розміщення інформаційних ресурсів на етапі проектування ІКС або АСК.

На етапі функціонування завдання розподілу ПМ, ІМ та їх відновного резервування вирішується при виході з ладу окремих компонентів системи й при введенні в експлуатацію нових ПМ. Для підвищення стійкості інформаційно-обчислювального процесу на етапі експлуатації мережі доцільно розподіл (перерозподіл) інформаційних ресурсів здійснювати за принципом «знизу-вгору».

Основним завданням при оптимізації інформаційно-обчислювального процесу є організація розподілу завдань за працездатними ЕОМ. Вирішення цього завдання пов'язане з характеристиками завдань і вимогами до виду деградації. При відмовах окремих ЕОМ можливий перерозподіл вирішуваних системою завдань між працездатними ЕОМ. Це дозволяє зберегти працездатність системи за рахунок зниження в допустимих межах яких-небудь показників якості її функціонування. Системи, у яких реалізується зазначена можливість, одержали назву систем з поступовою деградацією [4-15].

При виході з ладу значного числа ЕОМ контуру керування розподіл (перерозподіл) ПМ, ІМ та їх відновного резерву за працездатними ЕОМ виконується залежно від необхідності й доцільності вирішення завдань.

У контурах вищих рівнів розподіл програм, ІМ та їх резерву між контурами здійснюється з урахуванням мінімуму переданої інформації. У цьому випадку правильне і своєчасне вирішення зазначених завдань сприяє підтримці працездатності системи з колишньою продуктивністю та пропускною здатністю.

У межах проведених досліджень нами було розроблено комплекс взаємозалежних математичних моделей оптимізації відновного резервування інформації в ІКС або АСК, до складу якого входять: математична модель оптимізації відновного резервування інформації в ІКС або АСК і математична модель визначення параметрів оновлення цього резерву інформації.

Дані моделі дозволяють поетапно й взаємозалежно вирішувати завдання розподілу інформаційних ресурсів за вузлами мережі, їх відновного резервування.

Нехай задана мережа ІКС або АСК (контур керування), що складається з $L_{\text{еом}}$ ЕОМ, кожна з яких має im^*_j $(j=1,\ldots, L_{еом})$ пунктів опрацю-вання інформації (персональних ЕОМ, систем спостереження, GPS і т. п.).

У мережі вирішується K_{zad} завдань, які використовують дані з M_{inf} інформаційних масивів (ІМ). На кожному $h_{аб}$ -*м* пункті (абонентові) *j-й* ЕОМ $(j=1, 2, \ldots, L_{еом})$ $(h_{аб}=1, 2, \ldots, им^*_j)$ вирішується строго певне коло завдань із використанням певних ІМ і генерацією відповідних запитів (повідомлень).

Розподіл ПМ та ІМ за вузлами мережі визначається планом розподілу, що задається матрицями:

$$XM = \left\| хим^*_{kj} \right\|,\ YM = \left\| уим^*_{fj} \right\|,\ \Psi M = \left\| \psi им^*_{kj} \right\|,\ \Phi M = \left\| \varphi им^*_{fj} \right\|, \qquad (5.3)$$

де

$$хим^*_{kj} = \begin{cases} 1, \text{ якщо } k_{ПМ} - \text{й ПМ розміщено на } j - \text{й ЕОМ}, \\ 0, \text{ в протилежному випадку}, \end{cases} \qquad (5.4)$$

$$уим^*_{fj} = \begin{cases} 1, \text{ якщо } f_{ІМ} - \text{й ІМ розмішено на } j - \text{й ЕОМ}, \\ 0, \text{ в протилежному випадку}, \end{cases} \qquad (5.5)$$

$$\psi им^*_{kj} = \begin{cases} 1, \text{ якщо резерв } k_{ПМ} - \text{го ПМ розміщено на } j - \text{й ЕОМ}, \\ 0, \text{ в протилежному випадку}, \end{cases} \qquad (5.6)$$

$$\varphi им^*_{fj} = \begin{cases} 1, \text{ якщо резерв } f_{ІМ} - \text{го ІМ розміщено на } j - \text{й ЕОМ}, \\ 0, \text{ в протилежному випадку}, \end{cases} \qquad (5.7)$$

$k_{ПМ} = 1, 2, \ldots, K_{зад}$, $f_{ІМ} = 1, 2, \ldots, M_{\text{inf}}$, $j = 1, 2, \ldots, L_{еом}$.

Позначимо через rz_k, rz_f обсяг відновного резерву $k_{ПМ}$ *-го* ПМ і $f_{ІМ}$ *-го* ІМ (число копій (передісторій) $k_{ПМ}$ *-го* ПМ, $f_{ІМ}$ *-го* ІМ) *(*$k_{ПМ}$ *=1,2,...,*$K_{зад}$*,* $f_{ІМ}$ *=1,2,...,* M_{inf}*)* відповідно.

При постановці завдань оптимізації відновного резервування інформації в критично важливих АСК або ІКС можуть бути використані такі критерії: максимум ймовірності вирішення всіх завдань; мінімум часу на вирішення всіх завдань; мінімум обсягу інформації, що циркулює в мережі.

У результаті вирішення кожного завдання необхідно визначити підмножину вузлів ІКС або АСК, розміщення в кожному з яких ПМ (ІМ) та їх резерву забезпечує екстремальне значення використовуваного критерію оптимізації. Крім того, при вирішенні завдань оптимізації відновного резервування за критеріями максимуму ймовірності вирішення всіх завдань і мінімуму часу на їх рішення, необхідно визначити обсяг резерву.

За критерієм максимуму ймовірності вирішення всіх завдань, завдання оптимізації відновного резервування формулюється так.

Визначити значення $xum^*_{kj}, yum^*_{fj}, \psi um^*_{kj}, \varphi um^*_{fj}, rz_k, rz_f$ ($k_{ПМ}$=1,2,..., $K_{зад}$; j=1,2,..., $L_{еом}$; $f_{ІМ}$=1,2,..., M_{inf}) такі, що

$$P^{task} = max \prod_{j=1}^{L_{эвм}} \prod_{h_{аб}=1}^{um^*_j} \prod_{k_{ПМ}=1}^{k_{ПМ}} P_{j\,h\,k}(xum^*, yum^*, \psi um^*, \varphi um^*, rz_k, rz_f) \quad (5.8)$$

при обмеженнях:

а) на час вирішення $k_{ПМ}$ *-го* завдання $h_{аб}$ *-м* абонентом *j-ї* ЕОМ (АРМ, МАРМ)

$$T^{реш}_{j\,h\,k} \le T^{доп}_{j\,h\,k}, j=1,2,...,L_{еом}; h_{аб}=1,2,...,um^*_j; k_{аб}=1,2,...,M_{\text{inf}};$$

б) на обсяг інформації, що циркулює в мережі ІКС, при вирішенні $h_{аб}$ *-м* абонентом j *-го* вузла $k_{ПМ}$ *-го* завдання

$$\Lambda_{jhk} \le \Lambda_{jhk}^{don},\ j = 1,2,...,L_{еом};\ h_{аб} = 1,2,...,um^*_j;\ k_{ПМ} = 1,2,...,M_{\inf};$$

в) на обсяг зовнішнього запам'ятовувального пристрою j -ї ЕОМ

$$\sum_{k_{ПМ}=1}^{K_{зад}} (xum^*_{kj} + yum^*_{kj} rz_k) u_k +$$
$$\sum_{f_{ІМ}=1}^{M_{inf}} (yum^*_{fj} + \varphi um^*_{fj} \cdot rz_f)\delta_f \le V_j,\ j = 1,2,...,\ L_{еом};$$

г) на значення змінних

$$\sum_{j=1}^{L_{еом}} xum^*_{kj} = 1,\ \sum_{j=1}^{L_{еом}} yum^*_{gj} = 1,$$

$$g = 1,2,...,M_{\inf};\ j = 1,2,...,L_{еом};\ k_{ПМ} = 1,2,...,K_{зад};$$

$$xum^*_{kj} = \{0,1\};\ yum^*_{gj} = \{0,1\};$$

$$\sum_{j=1}^{L_{еом}} \psi um^*_{kj} = 1, \qquad \sum_{j=1}^{L_{еом}} \varphi um^*_{gj} = 1,$$

$$\psi um^*_{kj} = \{0,1\};\ \varphi um^*_{gj} = \{0,1\};$$

$$xum^*_{kj} + \psi um^*_{kj} < 2,\ yum^*_{gj} + \varphi um^*_{gj} < 2;$$

$$rz_k, rz_f = (0,1,2,3,...)(f_{ПМ} = 1,2,...,M_{\inf}),$$

де V_j – обсяг зовнішнього запам'ятовувального пристрою j -ї ЕОМ,

тоді одержимо наступні залежності для визначення ймовірності того, що ІМ не буде зруйнований і час вирішення завдання на вузлі [4-15]:

$$P_{jhk} = \tau_{jhk} \cdot \sum_{l=1}^{L_{еом}} P^P_{jhkl} \cdot P^n_{jlhk} \cdot xum^*_{kl} \cdot \prod_{f_{ПМ}=1}^{M_{inf}} \sum_{r=1}^{L_{еом}} yum^*_{fr} (P^0_{lkfr} \cdot \overline{P}^n_{lrkf})^{q_{jhkf}}, \qquad (5.9)$$

$$T_{ihk}^{реш} = \frac{1}{\tau_{jhk}} \sum_{l=1}^{L_{еом}} хит^{*}{}_{kl} \left(T_{jlhk} + t_{jhkl}^{реш} + Q_{jhkl}\, t_{kl}^{в} + \right.$$

$$\left. + \sum_{f_{ПМ}=1}^{M_{\inf}} q_{jhkf} \sum_{r=1}^{L_{еом}} \left(\overline{T}_{lrkf} + \overline{Q}_{lkfr}\, \bar{t}_{fr}^{\,в} \right) уит^{*}{}_{fr} \right., \qquad (5.10)$$

$$\Lambda_{jhk} = \lambda_{jhk} \left(\sum_{i=1}^{L_{еом}} \left\{ \begin{array}{l} хит^{*}{}_{ki} \left[\begin{array}{l} PA_{ji} l_{jhk}^{з} + PA_{ij} l_{jhk}^{с} + \\ + Q_{jhki} \sum_{l=1}^{L_{еом}} \psi_{kl} \left(PA_{il} l_{k}^{в} + PA_{li} uv_{k} \right) \end{array} \right] + \\ + \sum_{f_{ПМ}=1}^{M_{\inf}} \sum_{r=1}^{L_{еом}} \sum_{g=1}^{q_{jhkf}} уит_{fr} \left[\begin{array}{l} PA_{ri} \bar{l}_{kgf}^{з} + PA_{ir} \bar{l}_{kgf}^{с} + \\ + \overline{Q}_{ikfr} \sum_{l=1}^{L_{еом}} фит^{*}{}_{fl} [(PA_{rl} \bar{l}_{f}^{в} \\ + PA_{lr} \delta v_{k})] \end{array} \right] \end{array} \right\} \right), \qquad (5.11)$$

$T_{jhk}^{доп}$ – максимально припустимий час вирішення $h_{аб}$ - *м* абонентом *j-ї* ЕОМ ІКС та АСК $k_{ПМ}$ *-го* завдання;

$\Lambda_{jhk}^{доп}$ – максимально припустимий обсяг інформації, що циркулює в системі при вирішенні $h_{аб}$ *-м* абонентом *j -ї* ЕОМ $k_{ПМ}$ *-го* завдання;

P_{jihk}^{n} ($\overline{P}_{lrkf}^{\,n}$) – ймовірність успішної передачі інформації між вузлами *j (l)* і *i (r)* при вирішенні $h_{аб}$ *-м* абонентом (зверненні $k_{ПМ}$ *-го* ПМ) *j*-ї ЕОМ (розміщеного в l *-му* вузлі) $k_{ПМ}$ *-го* завдання (до $f_{ІМ}$ *-го* ІМ);

$P_{ijhk}^{з}$ ($\overline{P}_{lrkf}^{\,з}$), $P_{ijhk}^{с}$ ($\overline{P}_{rlkf}^{\,с}$) – ймовірності доведення запиту на вирішення (на доступ до інформації) і повідомлення, що містить результати рішення (звернення) $h_{аб}$ *-м* абонентом ($k_{ПМ}$ *-м* ПМ) *j -ї* ЕОМ (розміщеного в *l-й* ЕОМ) $k_{ПМ}$ *-го* завдання (до $f_{ІМ}$ *-го* ІМ) в *I -му* вузлі (що перебуває в *r-му* вузлі) ІКС або АСК відповідно;

P_{jhkl}^{p}, P_{lkfr}^{0} – ймовірність того, що $k_{ПМ}$ *-й* ПМ, що зберігається на *l-й* ЕОМ, не буде в процесі звернення до нього $h_{аб}$ *-м* абонентом *j-ї* ЕОМ зруйнований або ж буде успішно відновлений, і ймовірність того, що $f_{ІМ}$ *-й* ІМ, що зберігається на *r-й* ЕОМ, не буде в процесі звернення до нього $k_{ПМ}$ *-го* ПМ, що перебуває на *l-й* ЕОМ, зруйнований або ж буде успішно відновлений відповідно;

Q_{jhkl} ($\overline{Q}_{lkfr}$) – ймовірність того, що $k_{ПМ}$ -*й* ПМ ($f_{ІМ}$ - *ї* ІМ), що зберігається в *l* (*r*) -*му* вузлі буде зруйнований до моменту звернення до нього *h* -*го* абонента ($k_{ПМ}$ -*го* ПМ) *j* (*l*) -*ї* ЕОМ відповідно;

t^{B}_{kj}, ($\bar{t}^{B}_{fr}$) – середній час відновлення $k_{ПМ}$ -*го* ПМ ($f_{ІМ}$ -*го* ІМ) в *j*(*r*)-*м* вузлі;

T_{ijhk} ($\overline{T}_{lrkf}$) – середній час передачі повідомлення з *i*-*го* (*l*-*го*) вузла мережі в *j*-*й* (*r* - *й*) при вирішенні (зверненні) $h_{аб}$ -*м* абонентом ($k_{ПМ}$ -го ПМ) *j*-*й* ЕОМ (розміщеного в *l* -*м* вузлі) $k_{ПМ}$ -*го* завдання (до $f_{ІМ}$ - *му* ІМ);

q_{jhkf} – число звернень $k_{ПМ}$ -го ПМ до $f_{ІМ}$ -му ІМ при його вирішенні $h_{аб}$ -*м* абонентом *j* -*ї* ЕОМ;

$t^{реш}_{jhkl}$ – час вирішення $k_{ПМ}$ -*го* ПМ на *l*-*й* ЕОМ $h_{аб}$ -*м* абонентом *j*-*го* вузла при наявності всіх вихідних даних; $\tau_{jhk} = 1$, якщо $h_{аб}$ -*й* абонент *j*-*ї* ЕОМ має право вирішувати $k_{ПМ}$ -*е* завдання, $\tau_{jhk} = 0$ – в протилежному випадку;

λ_{jhk} – інтенсивність вирішення $k_{ПМ}$ -*го* завдання $h_{аб}$ -*м* абонентом *j*-*ї* ЕОМ;

$l^{З}_{jhk}$ ($\bar{l}^{З}_{kgf}$) – довжина запиту на вирішення $k_{ПМ}$ -*го* завдання ($f_{ІМ}$ -*му* ІМ) $h_{аб}$ -*м* абонентом ($k_{ПМ}$ -*м* ПМ) *j* -*ї* ЕОМ (при *g* -*м* зверненні до нього);

$l^{С}_{jhk}$ ($\bar{l}^{С}_{kgf}$) – довжина повідомлення, одержуваного в результаті вирішення (доступу до) $k_{ПМ}$ -*го* ПМ ($f_{ІМ}$ -*м* ІМ) $h_{аб}$ -м абонентом ($k_{ПМ}$ -*м* ПМ) *j*-ї ЕОМ (при g -*м* зверненні до нього);

$l^{В}_{k}$ ($\bar{l}^{В}_{f}$) – довжина запиту на відновлення $k_{ПМ}$ -*го* ПМ ($f_{ІМ}$ -*го* ІМ);

uv_k – обсяг $k_{ПМ}$ -*го* ПМ ІКС або АСК;

δv_f – обсяг $f_{ІМ}$ -*го* ІМ ІКС або АСК.

Більша розмірність загального завдання оптимізації відновного резервування інформації, дискретність, нелінійний характер цільових функцій і обмежень не дозволяє вирішувати його існуючими методами й висуває проблему зниження розмірності ПЗ та ІМ ІКС [4-15].

Для скорочення розмірності завдань оптимізації відновного резервування інформації, мабуть, необхідно виконати їх декомпозицію на ряд взаємозалежних підзавдань, які зведені до завдань таких класів [4-15]:

1) оптимізація розподілу ПМ і ІМ у системі обчислювальних засобів ІКС та АСК без урахування їх резервування - до класу завдань цілочисельного лінійного програмування зі змішаними обмеженнями;

2) оптимізація розподілу відновного резерву ПМ і ІМ без урахування можливості його руйнування (без визначення обсягу резерву) – до класу цілочисельних лінійних завдань;

3) оптимізація обсягу відновного резерву ПМ і ІМ – до двох стандартних завдань оптимального резервування.

Для вирішення завдань розподілу ПМ, ІМ і їх резерву по вузлах мережі пропонується використовувати метод віток і меж, а для вирішення завдання оптимізації обсягу відновного резерву – метод зустрічного вирішення функціональних рівнянь динамічного програмування [4-15].

Повне резервування стосується всієї системи й усіх файли. Повне резервування необхідно проводити, принаймні, щотижня. Даний рівень резервування забезпечується наступною стратегією (див. рис. 5.1).

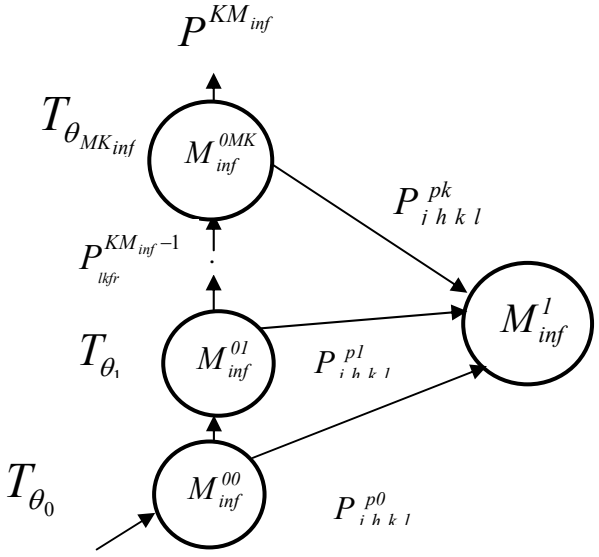


Рис. 5.1. Функціонування системи при використанні Стратегії 1

Основний масив M_{inf}^{00} резервується KM_{inf}^{k} копіями $M^{ork}{}_{inf}$, $rk = \overline{1, KM_{inf}^{k}}$. Ймовірність того, що масив $M^{ork}{}_{inf}$ не зруйнується за інтервал часу T_{θ_r} його використання, є $P^{p}{}_{j\,h\,k\,l}$. Ймовірність руйнування масиву в процесі відновлення $P^{0}{}_{lkfr} = 1 - P_{jhkl}^{p}$. Ймовірність руйнування ІМ під час їх зберігання – $P_{jhkl}^{p} = 0$. Ймовірнісний процес функціонування системи при вирішенні завдання відновлення може бути представлений у вигляді:

$$\sum_{j=1}^{KM_{inf}^{K}+1} \prod_{s=0}^{j} P^{0}{}_{lkfr(s)} - P^{p}{}_{jhkl(j_k)} = 1. \qquad (5.12)$$

Ймовірність успішного виконання завдання $P_{j\,h\,k\,l(1)}^{p}$ при наявності KM_{inf}^{k} копій дорівнює [4-15]:

$$P^{p}{}_{j\,h\,k\,l(1)} = 1 - \prod_{mx=0}^{KM_{inf}^{k}} P^{0}{}_{lkfr(mx)}, \qquad (5.13)$$

де індекс *"1"* позначає використання першої стратегії резервування.

Нехай $T_{rez(j)}$ – час створення j_k -*ї* копії основного масиву, $j_k = \overline{1, KM_{inf}^{k}}$. Тоді планований середній час доступу до ЕОМ у складі та ІКС (АРМ у АСК) при використанні стратегії 1 визначиться, враховуючи роботи [4-15], так:

$$M(T_{d1}) = \sum_{j_k}^{KM_{inf}^{k}} \Big(\sum_{i=0}^{j_k} T_{\theta_i} \prod_{s=0}^{i=1} P^{0}{}_{lkfr(s)} - P^{p}{}_{jhkl(j_k)} + \sum_{n=0}^{KM_{inf}^{k}} P^{0}{}_{lkfr(mx)} + P^{p}{}_{j\,h\,k\,l(1)} \cdot \sum_{j_k=0}^{KM_{inf}^{k}} T_{rez(j)} \Big). \qquad (5.14)$$

Отриманий вираз можна перетворити на більш компактний, якщо врахувати, що $\prod_{j_k=0}^{-1} P^{0}{}_{lkfr(j_k)} = 1$:

$$M(T_{dl}) = \sum_{i=0}^{KM_{inf}^{k}} T_{\theta_r} \prod_{j_k=0}^{i-1} P^{0}{}_{lkfr(j_k)} + P^{p}{}_{jhkl(1)} \sum_{j_k=1}^{KM_{inf}^{k}} T_{rez(j)}. \qquad (5.15)$$

При диференціальному резервуванні кожен файл, який був змінений з моменту останнього повного резервування, копіюється кожен раз заново. Диференціальне резервування прискорює процес відновлення. Все, що необхідно, це остання повна й остання диференціальна резервна копія. Даний рівень резервування забезпечується стратегією, яка може бути змодельована так.

Стратегія 2. Нехай у початковий момент в системі є k - передісторії і масивів змін основного масиву $\mathrm{M}_{\mathrm{inf}}^{00}$, тобто $\mathrm{M}_{\mathrm{inf}}^{-1}$, $\mathrm{M}_{\mathrm{inf}}^{-2}$,..., $\mathrm{M}_{\mathrm{inf}}^{-KM_{\mathrm{inf}}^{k}}$, які використовуються для розв'язання деякої задачі. Потрібно створити оновлений масив. За фіксований (одиничний) час з ймовірністю P_{jhkl}^{p} оновлений масив може бути створений, а з ймовірністю $\mathrm{P}^{0}{}_{\mathrm{lkfr}} = 1 - \mathrm{P}_{\mathrm{jhkl}}^{\mathrm{p}}$ масив може бути зруйнований. Робота системи продовжується доти, доки або масив $\mathrm{M}_{\mathrm{inf}}^{00}$ і $KM_{\mathrm{inf}}^{\mathrm{p}k}$ його передісторії не будуть зруйновані, або не буде створено оновлений масив.

Визначимо такі характеристики системи при використанні даної стратегії резервування: ймовірність поновлення масиву $\mathrm{M}_{\mathrm{inf}}^{00}$ ймовірність руйнування масиву і $KM_{\mathrm{inf}}^{\mathrm{p}k}$ його передісторії; тривалість роботи ЕОМ до поновлення масиву, до руйнації всіх передісторії, а також середній час функціонування ЕОМ.

Будемо інтерпретувати результати оновлення масивів як переміщення точки по осі Z_{os}. У момент $\mathrm{T}_{\mathrm{I}} = 0$ точка знаходиться в положенні $Z_{os} = KM_{\mathrm{inf}}^{pk} + 1$, а в моменти $\mathrm{T}_{\mathrm{I}} = 1,2...$ вона переміщується на один крок вліво чи вправо залежно від успіху чи невдачі відповідного оновлення, тобто здійснює випадковий рух. Таким чином, положення точки в довільний момент

$T_{I\xi}$ визначає кількість невирішених передісторій після закінчення $T_{I\xi}$ -*ї* спроби поновлення.

Розглянемо часові характеристики процесу при використанні стратегії 2. Для визначення середньої тривалості роботи ЕОМ до поновлення масиву $\mathrm{M}_{\mathrm{inf}}^{00}$ і до руйнування цього масиву і KM_{inf}^{pk} його передісторії застосуємо метод твірних функцій. Дане завдання аналогічне до завдання визначення тривалості випадкового руху точки з початкового стану при наявності поглинаючих екранів при $Z_{os} = KM_{\mathrm{inf}}^{pk} + 2$ і $Z_{os} = 0$. Так само, як і раніше, вихідним положенням точки є $Z_{os}(0 < Z_{os} < KM_{\mathrm{inf}}^{pk} + 2)$.

Нехай $PU_{z_{os},T_{I\zeta}}$ - ймовірність того, що процес закінчиться на $T_{I\xi}$ -*м* кроці ($Z_{os} = 0$).

Після першої спроби поновлення система потрапляє або в точку $Z_{os} + 1$ або в точку $Z_{os} - 1$, і при $1 < Z_{os} < KM_{\mathrm{inf}}^{pk} + 1$ і $\mathrm{T}_{\mathrm{I}\zeta} > 1$ одержимо:

$$PU_{z_{os},T_{I\zeta}+1} = P^{p}_{j\,h\,k\,l} \cdot PU_{z_{os}+1,x_{os}} + P^{0}{}_{lkfr} \cdot U_{z_{os}-1,T_{I\zeta}}\ .$$

Дане рівняння є різницевим і залежить від двох змінних.

Граничні умови мають вигляд: $PU_{Z_{os},T_{I\zeta}} = PU_{KM_{\mathrm{inf}}^{pk}+2,T_{I\zeta}} = 0$ при $T_{I\zeta} \geq 1$ и $PU^{00} = 1, PU_{Z_{os},0}$ при $Z_{os} > 0, 0 < Z_{os} < KM_{inf}^{pk} + 2$, і $T_{I\zeta} \geq 0.$ Введемо твірну функцію:

$$PU_{Z_{os}(S)} = \sum_{T_{I\zeta}=0}^{\infty} PU_{Z_{os},T_{I\zeta}} \cdot S^{T_{I\zeta}}.$$

Ця твірна функція ймовірності руйнування масиву $\mathrm{M}_{\mathrm{inf}}^{00}$ і всіх його передісторії на $T_{I\xi}$ -*му* кроці може бути представлена у вигляді:

$$PU_{l(S)} = (P^{0}{}_{lkfr} \cdot (P^{p}_{jhkl})^{-1})^{Z_{os}} \cdot$$
$$\cdot [\lambda_1^{KM^{pk}_{inf}+2}(s) - \lambda_2^{KM^{pk}_{inf}+2}(s)], \qquad (5.16)$$

де

$$\lambda_1(s) = (2 \cdot P^{p}_{jhkl} \cdot s)^{-1}[1 + (1 - 4 \cdot P^{p}_{jhkl} \cdot P^{0}{}_{lkfr} \cdot s^2)^{0.5}];$$
$$\lambda_2(s) = (2 \cdot P^{p}_{jhkl} \cdot s)^{-1}[1 - (1 - 4 \cdot P^{p}_{jhkl} \cdot P^{0}{}_{lkfr} \cdot s^2)^{0.5}],$$

для $0 < s < 1$.

Тоді математичне очікування часу до руйнування масиву і всіх його передісторії $M(T_{d,s})$ за умови, що процес функціонування почався в точці Z_{os}, може бути визначений як $dPU_{z_{os}}(s)/ds|_{s\to 1} = M(T_{d,s})$. Можна показати, що для $Z_{os} = KM^{pk}_{inf} + 1$ маємо:

$$M(T_{d,KM^{pk}_{inf}+1}) = (P^{p}_{jhkl} \cdot (P^{0}{}_{lkfr})^{-1})^{KM^{pk}_{inf}+1} \cdot CK\,(AK^{KM^{pk}_{inf}+2} -$$
$$- BK^{KM^{pk}_{inf}+2})^{-2} \cdot DK,$$
$$DK = (KM^{pk}_{inf} + 2) \cdot (AK^{KM^{pk}_{inf}+2} + BK^{KM^{pk}_{inf}+2}) \cdot (AK - BK) -$$
$$- (AK^{KM^{pk}_{inf}+2} - BK^{KM^{pk}_{inf}+2}) \cdot (AK - BK);$$
$$AK = \lambda_1(s)|_{s\to 1}; \qquad (5.17)$$
$$BK = \lambda_2(s)|_{s\to 1};$$
$$CK = (1 - 4 \cdot P^{p}_{jhkl} \cdot P^{0}{}_{lkfr})^{0.5}.$$

Твірна функція ймовірності успішного оновлення масиву $PF_{z_{os}}(s)$ (успішного виконання завдання) на $T_{I\xi}$ - *му* кроці має вигляд:

$$PF_{z_{os}}(s) = (P^{p}_{j\,h\,k\,l} \cdot (P^{0}{}_{lkfr})^{-1})^{KM^{pk}_{inf}+2-z_{os}} \cdot$$
$$\cdot [\lambda_1^{KM^{pk}_{inf}+2}(s) - \lambda_2^{KM^{pk}_{inf}+2}(s)]^{-1} \cdot \qquad (5.18)$$
$$\cdot [\lambda_1^{z_{os}}(s) - \lambda_2^{z_{os}}(s)].$$

Математичне очікування часу до успішного закінчення виконання завдання $\mathrm{M}(\mathrm{T}_{\mathrm{P}^0{}_{\mathrm{lkfr}},z_{os}})$ за умови, що процес почався в точці Z_{os}, визначимо як $\mathrm{dPF}_{z_{os}}(s)/ds\,|_{s\to 1} = M(T_{\mathrm{P}^0{}_{\mathrm{lkfr}},z})$. Можна показати, що при $Z_{os} = KM^{pk}_{\mathrm{inf}} + 1$ одержимо:

$$M[T_2^{y}] = M(T_{P^0{}_{lkfr},KM^{pk}_{inf}+1}) = (P^{p}_{j\,h\,k\,l} \cdot (P^{0}{}_{lkfr})^{-1}) \cdot CK(AK^{KM^{pk}_{inf}+2} -$$
$$- BK^{KM^{pk}_{inf}+2})^{-2} \cdot DK,$$
$$DK = (KM^{pk}_{inf} + 2) \cdot (AK^{KM^{pk}_{inf}+2} + BK^{KM^{pk}_{inf}+2}) \cdot (AK^{KM^{pk}_{inf}+1} - BK^{KM^{pk}_{inf}+1})$$
$$- (KM^{pk}_{inf} + 1) \cdot (AK^{KM^{pk}_{inf}+2} + BK^{KM^{pk}_{inf}+2})(AK^{KM^{pk}_{inf}+1} - BK^{KM^{pk}_{inf}+1}); \qquad (5.19)$$
$$AK = \lambda_1(s)\,|_{s\to 1};$$
$$BK = \lambda_2(s)\,|_{s\to 1};$$
$$CK = (1 - 4 \cdot P^{p}_{j\,h\,k\,l} \cdot P^{0}{}_{lkfr})^{0.5}.$$

Твірною функцією тривалості функціонування ЕОМ ІКС та АСК до поновлення масиву $\mathrm{M}^{00}_{\mathrm{inf}}$, або до руйнування всіх масивів буде сума твірних функцій (5.16) і (5.18). Однак якщо тривалість функціонування має кінцеве математичне очікування, $\mathrm{P}^{\mathrm{p}}_{\mathrm{j\,h\,k\,l}} \neq \mathrm{P}^{0}{}_{\mathrm{lkfr}}$, то для визначення середньої тривалості функціонування ЕОМ можна застосувати більш простий метод. Міркуючи так само, як при виведенні рівняння (5.16), отримаємо $\mathrm{PD}_{z_{os}} = \mathrm{P}^{0}_{\mathrm{j\,h\,k\,l}} \cdot PD_{z_{os}+1} + \mathrm{P}^{\mathrm{p}}_{\mathrm{j\,h\,k\,l}} \cdot PD_{z_{os}-1} + 1$ при $0 < Z_{os} < KM^{pk}_{\mathrm{inf}} + 2$ і граничні умови $PD_0 = 0, \quad PD_{KM^{pk}_{\mathrm{inf}}+2} = 0$.

Враховуючи, що $Z_{os} = KM^{pk}_{inf} + 1$ та $P^{0}{}_{lkfr} < P^{p}_{j\,h\,k\,l}$, одержимо:

$$M\ [\ T_2\ ]\ =\ T_{\theta_i}\cdot PD_{\ KM\ _{inf}^{pk}+1}\ =\ [\ \frac{T_{\theta_i}}{P^{\,0}{}_{lkfr}\ -\ P^{\,p}_{j\ h\ k\ l}}\ ]\cdot[\ KM\ _{inf}^{pk}\ +1\ ].\qquad(5.20)$$

При додатковому резервуванні відбувається копіювання тільки тих файлів, які були змінені з того часу, як востаннє виконувалося повне або додаткове резервне копіювання. Подальше додаткове резервування додає тільки файли, які були змінені з моменту попереднього додаткового резервування. У середньому додаткове резервування займає менше часу через те, що копіюється менша кількість файлів. Проте процес відновлення даних займає більше часу, тому що повинні бути відновлені дані останнього повного резервування, а також дані всіх наступних додаткових резервувань.

Розглянута модель розрахунку характеристик стратегій резервування ІМ ІКС та АСК дозволяє проводити аналіз практично будь-яких систем резервування ПМ и ІМ. Крім того, дана узагальнена модель забезпечує можливість аналізу і вибору оптимальних за часовим, ймовірнісним або вартісним критеріями варіантів розміщення резервних копій і (або) передісторії на носіях різного типу, а також враховує особливості роботи з ієрархічною пам'яттю.

5.2. Оптимізація завдань захисту інформаційно-комунікаційного середовища

Особливістю сформульованих завдань оптимізації складу комплексів ЗЗІ та відновного резервування інформації є наявність обмежень, здійсненність яких перевіряється аналітичними методами або методом імітаційного моделювання. Для вирішення завдань такого класу запропоновано два підходи:

1) включення в схему розгалуження нелінійних обмежень;

2) вирішення скороченого завдання без урахування нелінійних обмежень, і на отриманій множині допустимих рішень перевірка здійсненності цих обмежень.

При виконанні завдання із використанням першого підходу, на наш погляд, доцільно використовувати алгоритми, що ґрунтуються на ідеях методу віток і меж. Для визначення множини допустимих рішень скороченого завдання нами запропонований модифікований метод зустрічного рішення функціональних рівнянь динамічного програмування. При модифікації методу нами було враховано показники, що характеризують відношення власника інформації до потенційних ризиків, пов'язаних з інвестуванням у засоби захисту інформації, а саме варіанти, при яких власник може обирати з наступних варіантів:

1) захист всіх вузлів ІКС та АСК;

2) вибірковий захист тільки тих вузлів ІКС, які мають сполучення з МЗК (як найбільш уразливих);

3) захист вузлів з критично важливою інформацією для бізнес-процесів компанії;

4) інше.

Для синтезу підсистем захисту необхідно обрати таку їх сукупність, яка забезпечує або мінімальну ймовірність НСД доступу при обмеженнях на вартісні, часові та інші показники, або мінімальні сумарні втрати від подолання захисту та витрат на розробку й експлуатацію ЗЗІ, або мінімальні витрати на розробку й експлуатацію системи при обмеженнях на ймовірність подолання захисту.

На загальну ефективність застосування методу віток і меж для вирішення завдання оптимізації розподілу ПМ та ІМ, а також їх відновного резерву впливає вибір стратегії розгалуження та методу оцінки меж рішення. Пропонується для скорочення обчислювальної складності методу віток і меж оцінку меж рішення здійснювати наближеним методом рішення двоїстого, стосовно початкового, завдання, із застосуванням теорії двоїстості для попереднього визначення порядку розгалуження змінних. Це дозволить, при незначному погіршенні точності визначення меж рішення, скоротити загальний час вирішення завдань за рахунок меншої, порівняно з точними методами,

обчислювальної складності визначення меж рішення даними методами [4-15 та ін.].

У загальному випадку розроблені завдання оптимізації розподілу ПМ, ІМ та їх відновного резерву можуть бути приведені, з урахуванням запропонованих вище підходів, а також з урахуванням робіт [4-15], до такого вигляду.

Визначити такі складові вектора рішення $X=(x_1,x_2,...,x_n)$, які максимізують функцію:

$$PA(x)=\sum_{j=1}^{n} c_j x_j \tag{5.21}$$

в зоні, заданій обмеженнями:

$$x_j = \{0,1\},\ j = 1,2,...,\ n \tag{5.22}$$

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i \quad i = 1,2,...,\ m. \tag{5.23}$$

Для оцінки меж рішення умова (5.22) послаблюється і замінюється умовою: $0 \le x_j \le 1 \qquad j = 1,n.$

Тоді двоїстою, стосовно задачі (5.21) і (5.22), є задача

$$ZL = min\left(\sum_{i=1}^{m} b_i y_i + \sum_{i=m+1}^{m+n} y_i\right), \tag{5.24}$$

при обмеженнях:

$$\sum_{i=1}^{m} a_{ij} y_i + y_{m+j} \ge c_j,\ \ j = 1,2,...,n, \tag{5.25}$$

$$y_i \ge 0, \quad i = 1,2,...,\ m+n. \tag{5.26}$$

У розділі 3 ми визначили умовну ймовірність подолання захисту як $\Phi_{i,j(t)}$ для $D_{сзи_i}$ засобу (методу) захисту, що належить до рубежу j і закріпленого за об'єктом p_a за умови подолання рівнів $j-1, j-2,...,1$.

Позначимо $Q_{matrix} = \{q_{ei}, e = 0,1,..., n, \quad i = 1,2,..., m+n\}$ – матриця, k-ий рядок якої становить рішення двоїстої задачі (5.24) - (5.26), але при $j = i_1, i_2, ..., i_k$; $KS = KS_1 \cup KS_0$ – множина індексів змінних, включених до KS – часткове розв'язання (тут $KS_1 = \{j \setminus x_j = 1\}$, $KS_0 = \{j \setminus x_j = 0\}$; $KU = \{j: j = 1,2,...n\}$ – множина індексів змінної основної задачі.

Тоді наближений алгоритм оцінки меж розв'язанням двоїстої задачі з визначенням порядку розгалуження змінних включає наступні кроки, описані в таблиці 5.1.

Таблиця 5.1

Алгоритм оцінки меж розв'язання двоїстої задачі з визначенням порядку розгалуження для вирішення завдань забезпечення захисту інформації в ІКС

№ кроку	Дія	Залежність для параметру, що розраховується	Примітка
1	Сформулювати завдання	$1 - \prod_{p_a \in PA} \prod_{j=1}^{j_{p_a}} \sum_{D_{сзи}=1}^{D_{сзи\,j}} \Phi_{i,j(t)}(p_a)$ $\cdot ws_{D_{cpbj}}(p_a) \to max,$	$ws_{D_{cpbj}}(p_a) = \begin{cases} 1, \text{ якщо } D_{сзз}, \\ \textit{яке належить до } j\text{-го } \textit{рівня}, \\ \textit{закріплено за об'єктом } p_a; \\ 0, \textit{ в протилежному випадку}. \end{cases}$

2	Визначити величину d_i^k	$d_i^k = \frac{b_i}{\sum_{j \in I^k} a_{ij}}$, $i = 1,2,..., m+n$, $b_{m+j} = 1,\ j = 1,2,...,n$	де $k = 1,2,...$ – номер ітерації, I^k – множина індексів умов (5.25), для яких нерівність не виконується $\left(I^1 = \{1,2,...,n\}\right)$
3	Вибрати змінну y_r^k	$d_r^k = \min_i d_i^k$	
4	Обчислити значення змінної y_r^k та індекс змінної для розгалуження на $(n-k-1)$ - ому ярусі дере-ва розгалужень	$y_r^k = \min_j \frac{AS}{a_{rj}}$, $AS = c_j -$ де $-\sum_{i=1}^{m+n}\sum_{t=1}^{k-1} a_{ij} y_i^t$, $y_i^0 \equiv 0$, $i = 1,2,..., m+n$	Індекс i_q, який визначає мінімум y_r^k, є індексом змінної x_{ik} для розгалуження на $(n-k+1)$-*ому* ярусі. Записати $P(k) = i_q$.
5	Визначити значення елементів *k*-го рядку матриці	$q_{ki} = q_{k-1,i} + y_r^k$, $i = 1,2,..., m+n$, де $q_{0i} \equiv 0$, $i = 1,2,..., m+n$.	
6	Виключити з множини I^k індекс рівняння, для якого	$\sum_{i=1}^{m} a_{ij} y_i + y_{m+j} \geq c_j$, $j = 1,2,..., n$	Перевірити умову $k = n$, якщо умова не виконується, то покласти $k=k+1$ і перейти до кроку 2, в іншому випадку - до кроку 7.

Продовження таблиці 5.1

7	Розрахувати y_i	$y_i = \sum_{j=1}^{k} y_i^{\,j}$, $(i = 1,2,...,m+n)$, $ZL = \sum_{i=1}^{m} b_i y_i + \sum_{i=m+1}^{m+n} y_i$	При розв'язанні задачі (5.22 – 5.23) порядок розгалуження змінних визначається масивом $PH = \{i_1, i_2, ..., i_n\}$. Перший елемент x_{i_n}, а потім $x_{i_{n-1}}$ і т. д.

Для обґрунтованості подібного вибору проведена експериментальна перевірка ефективності стратегій розгалуження та впливу точності оцінки меж рішення на ефективність методу віток і меж.

Для підвищення ефективності методу зустрічного рішення функціональних рівнянь динамічного програмування, при вирішенні завдання оптимізації складу комплексів ЗЗІ, а також обсягу відновного резерву ПМ та ІМ, пропонується використовувати спосіб упорядкування обмежень за жорсткістю на основі застосування теорії двоїстості.

Передбачається, що втрати можуть бути викликані не тільки фактом порушення безпеки об'єкта захисту p_a, але і проникненням за j-*й* рівень захисту шляхом подолання $D_{сзі_i}$-*го* засобу (методу) захисту. Таким чином, порушник отримує можливість користуватися інформацією, яка визначається рівнем j і $D_{сзі_i}$-*м* засобом, на шкоду системі. Якщо на j-*му* рівні захисту можливе одночасне використання декількох методів, то ці методи об'єднують в один і відповідним чином коригують ймовірнісні, часові та вартісні характеристики.

Вихідна задача, з урахуванням результатів робіт [4-15], представлена в такому вигляді. Потрібно максимізувати цільову функцію:

$$FR = \sum_{j=1}^{N} r_j(x_j), \tag{5.27}$$

при обмеженнях:

$$\sum_{j=1}^{N} d_{ij}(x_j) \le D_{\text{сзі}_i}, \qquad i = 1,2,.., \qquad MI, \tag{5.28}$$

$$x_j = 1,2,...A_j, \qquad j = 1,2,..., \ N, \tag{5.29}$$

де $d_{ij}(x_j) > 0, \quad r_j(x_j) > 0, \quad j = 1,2,...,N, \ i = 1,2,...,MI.$

На підставі принципу оптимальності методу динамічного програмування можна скласти два функціональних рівняння:

$$f_n(D_{\text{сзі}_{1n}}, D_{\text{сзі}_{2n}}, ..., D_{\text{сзі}_{mn}}) = \max_{x_{un} \in X_{un}} \left\{ f_{n-1} \left[\begin{array}{l} D_{\text{сзі}_{1n}} - d_{1n}(x_n), D_{\text{сзі}_{2n}} - \\ - d_{2n}(x_n), ..., D_{\text{сзі}_{mn}} - d_{mn}(x_n) \end{array} \right] + r_n(x_n) \right\}, \tag{5.30}$$

$$n = 1,2,..., \ N, \qquad m = 1,2,..., \ MI.$$

$$\begin{aligned} &PA_n(D^0_{\text{сзі}_{1n}}, D^0_{\text{сзі}_{2n}}, ..., D^0_{\text{сзі}_{mn}}) = \\ &= \max_{x_n \in X_n} \left\{ PA_{un+1} \left[\begin{array}{l} D^0_{\text{сзі}_{1n}} - d_{1n}(x_n), \\ D^0_{\text{сзі}_{2n}} - d_{2n}(x_n), ..., D^0_{\text{сзі}_{mn}} - d_{\text{сзі}_{mn}}(x_n) \end{array} \right] + \right. \\ &\left. + r_n(x_n) \right\}, \end{aligned} \tag{5.31}$$

$$n = N, N-1, ..., 1, \qquad m = 1,2,..., MI,$$

де $$D_{\text{сзі}_{in}} = \sum_{j=1}^{n} d_{ij}(x_j), \quad D^0_{\text{сзі}_{in}} = \sum_{j=n}^{N} d_{ij}(x_j), \ i = 1,2,...m.$$

Функціональні рівняння (5.30) і (5.31) відрізняються від звичайних функціональних рівнянь тим, що кількість обмежень у них не є постійною величиною, вони можуть бути розв'язані при різних значеннях $m = 1,2,..., MI.$

Для переходу до двоїстої задачі спільне завдання (5.27) – (5.29) подається у вигляді лінійної задачі з додатковими обмеженнями:

$$FR = max \sum_{j=1}^{N} \sum_{k=1}^{A_j} r_{kj} x_{kj},$$

$$\sum_{j=1}^{N} \sum_{k=1}^{A_j} d_{ikj} x_{kj} \le D_{\text{сзі}_i}, \qquad i = 1,2,..., MI \tag{5.32}$$

$$\sum_{k=1}^{A_j} x_{kj} = 1, \qquad j = 1,2,..., N. \tag{5.33}$$

$$x_{kj} = \{0,1\}, \quad k = 1,2,..., A_j, \quad j = 1,2,..., N$$

Умова (5.33) послаблюється і замінюється умовою

$$0 \le x_{kj} \le 1,$$

тоді двоїстою щодо задачі (5.32) є задача:

$$ZL = \min\left(\sum_{i=1}^{MI} D_{\text{сзі}_i} y_i + \sum_{i=MI+1}^{MI+N} y_i \right), \tag{5.34}$$

$$\sum_{i=1}^{MI} d_{ikj} y_i + y_{MI+j} \ge r_{kj}, \tag{5.35}$$

$$k = 1,2,..., A_j, \quad j = 1,2,..., N, \quad y_i \ge 0, \quad i = 1,2,..., MI + N. \tag{5.36}$$

Задача (5.34) – (5.36) використовується для впорядкування обмежень за жорсткістю.

З економічної інтерпретації двоїстої задачі випливає, що чим більше значення змінної, тим більш жорстким є відповідне їй обмеження. З огляду на це обмеження вихідної задачі необхідно розставити в порядку $i_1, i_2, ..., i_{MI}$, що задовольняє умові $y_{i_1} \geq y_{i_2} \geq ... \geq y_{i_{MI}}$.

Обчислювальний процес починається з розв'язання функціонального рівняння (5.30) при *m=1*. Обсяг необхідної пам'яті ЕОМ і час виконання завдання при використанні методу зустрічного розв'язання функціональних рівнянь динамічного програмування, в основному, визначається розв'язанням задачі за першим обмеженням.

Розв'язанням рівняння (5.30) визначаються оптимальні послідовності $f_n(D_{1n})$, відповідні їм залежності $x_n(f_n), r_n(x_n)$ $(n = N, N-1, ..., 1)$ і функції $D_{\text{сзі}_{iN}}(f_N)$ (i = 2,3,...,MI). Значення послідовностей $f_N(D_{\text{сзі}_{iN}}), D_{\text{сзі}_{iN}}(f_N)$ $(i = 1,2,..., MI)$ дозволяють визначити максимальне значення $f_N(D_{\text{сзі}_{iN}}) = FR_{11}$, при якому виконується обмеження (5.32) при *i=1*, і максимальне значення $f_n(D_{\text{сзі}_{in}}) = FR_{12}$, при якому виконується (5.32) при *i=1,2,...,MI*. Якщо $FR_{11} = FR_{12}$, то значення змінних x_n $(n=1,2,...,N)$, відповідних значенню FR_{11}, є рішенням задачі, в іншому випадку $(FR_{11} > FR_{12})$ здійснюється перехід до другої ітерації, яка полягає у розв'язанні функціонального рівняння (5.31) при *m*=2, і т. д.

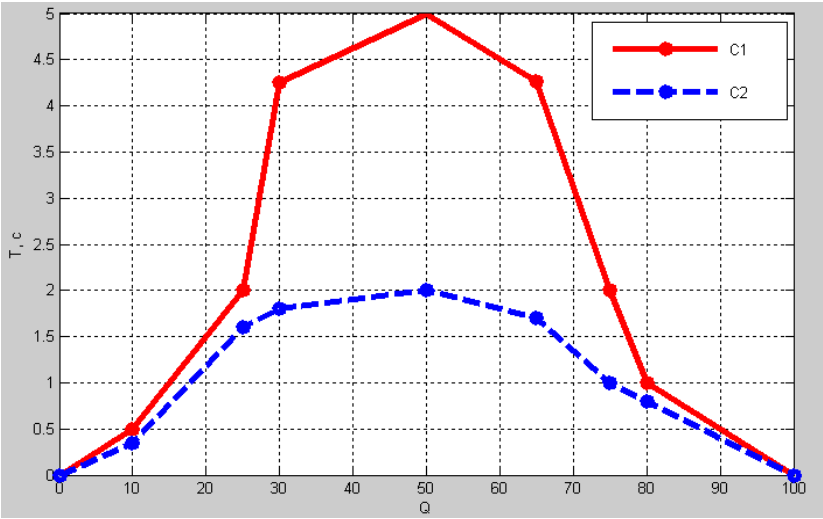
В ході перевірки математичної моделі оптимізації розподілу ПМ, ІМ та їх відновного резервування за вузлами ІКС було проведено розподіл ПМ та ІМ за критерієм мінімуму переданої інформації і розподіл відновного резерву за критерієм максимуму ймовірності вирішення всіх завдань ІС або АСК.

Ефективність методу віток і меж оцінювалася при розв'язанні задачі про ранці різної розмірності. Результати моделювання показали, що найбільш ефективними з розглянутих стратегій розгалуження є глобально-пошукова та

локально-виборча, з перевагою глобально-пошукової стратегії. Найменш ефективною є фронтальна стратегія розгалуження, яка при кількості змінних у розв'язуваних задачах вже вище за 20 поводиться вкрай нестабільно. При різних вихідних даних час розв’язання задач однієї і тієї ж розмірності при використанні фронтальної стратегії може відрізнятися в широких межах, тому що при деяких умовах дана стратегія вироджується (повністю або частково) в повний перебір варіантів рішення.

Варто відмітити стабільність такої поведінки стратегій розгалуження незалежно від рівня жорсткості обмежень, заповнювання матриці обмежень, а також від методу оцінки меж рішення. У разі жорсткого обмеження на обсяг оперативної пам'яті, займаної при вирішенні завдання, кращим є використання локально-виборчої стратегії розгалуження змінних, як такої, що найбільш раціонально використовує пам'ять ЕОМ.

Результати експериментальної перевірки ефективності впливу попереднього визначення порядку розгалуження змінних наведені на рис. 5.2.



C1 - симплекс-метод без визначення порядку розгалужень

C2 - симплекс-метод з визначенням порядку розгалужень

Рис. 5.2. Порівняльна характеристика методу віток і меж для різних значень жорсткості обмежень

Аналіз результатів показав, що застосування способу попереднього визначення порядку розгалуження змінних разом із симплекс-методом оцінки меж рішення дозволяє скоротити час вирішення завдань в 5–20 разів.

Аналіз отриманих результатів показав, що обсяг інформації, яка циркулює в системі, за рахунок раціонального розподілу ПМ та ІМ зменшився на 17–20%, при цьому більш ніж на 35% зросла ймовірність вирішення всіх завдань в системі при мінімальному обсязі відновного резерву кожного ПМ та ІМ, при одночасному збільшенні стійкості інформаційно-обчислювального процесу.

Результати моделювання показали, що ймовірність вирішення всіх завдань у системі з урахуванням відновного резервування при обсязі резерву в одну копію зросла на 28–30% при збільшенні ймовірності вирішення кожного із завдань не нижче ніж на 5–7%.

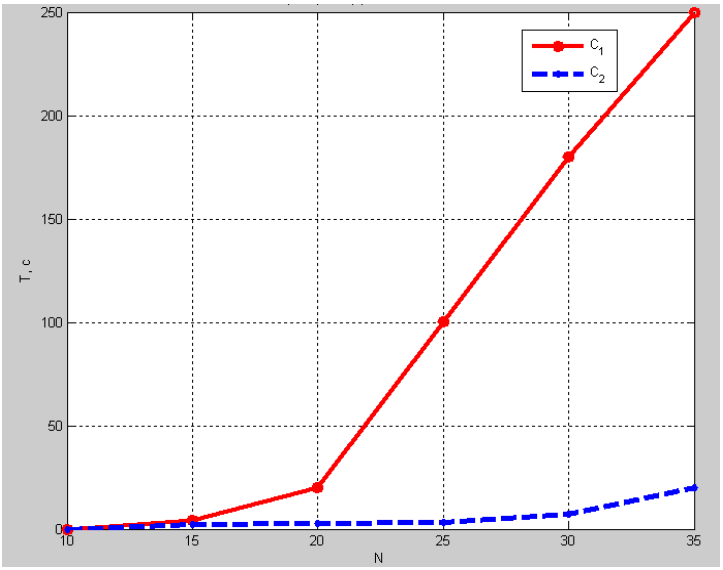
В ході моделювання виконано оцінку впливу відновного резервування на своєчасність вирішення завдань у системі. Як показав аналіз, середній час виконання завдання з урахуванням відновного резервування для забезпечення збереження інформації склав близько 3,2–3,4 с, а ймовірність вирішення за час, що не перевищує 7 секунд, – 0.97.

Дані результати підтвердили працездатність і доцільність використання розроблених методик і алгоритмів оптимізації відновного резервування інформації в ІКС та АСК для вирішення практичних завдань.

Отже, для вирішення завдань оптимізації розподілу ПМ, ІМ та їх відновного резерву за вузлами ІКС доцільно застосовувати метод віток і меж із використанням глобально-пошукової стратегії розгалуження змінних. У разі дефіциту оперативної пам'яті, як стратегії розгалуження, найбільш ефективною є локально-виборча стратегія. Як метод оцінки меж рішення може бути використаний наближений метод розв’язання подвійної задачі з попереднім визначенням порядку розгалуження змінних. Однак, виділення алгоритму визначення порядку розгалуження змінних в окрему процедуру і використання

її разом із симплекс – методом робить цей метод оцінки меж рішення кращим для використання.

Оцінка ефективності методу зустрічного розв'язання функціональних рівнянь, що використовує принцип подвійності для впорядкування обмежень за жорсткістю, проводилася за часом вирішення завдання вибору складу комплексу технічних засобів і числом ітерацій. Результати експерименту, представлені на рис. 5.3, показали, що попереднє впорядкування обмежень за жорсткістю дозволяє зменшити час вирішення від 10 до 60 разів. Це виходить за рахунок того, що більшість (до 80%) вирішуваних завдань вирішується за перші 2–3 ітерації за найбільш жорсткими обмеженнями, а також за рахунок значного відсіву безперспективних варіантів у ході перших ітерацій, що призводить до зниження числа членів умовно оптимальних послідовностей.



C_1 - без застосуванням теорії двоїстості;

C_2 - із застосуванням теорії двоїстості для впорядкування обмежень за жорсткістю

Рис. 5.3. Результати перевірки ефективності методу зустрічного рішення функціональних рівнянь динамічного програмування (M=5)

Виграш у часі вирішення розроблених завдань оптимізації складу комплексів засобів захисту та відновного резервування інформації тим більше важливий, якщо врахувати, що алгоритми та їх програмна реалізація частково

включаються до складу спеціального ПЗ і впливають на загальний час вирішення функціональних завдань.

Таким чином, використання розробленого комплексу моделей, методів і алгоритмів оптимізації інформаційно-обчислювального процесу і забезпечення схоронності й захищеності інформації дозволить підвищити обґрунтованість прийнятих рішень на етапах проектування ІКС, експлуатації та реконструкції ІКС з розподіленим опрацюванням.

Аналіз принципів функціонування системи, що захищається, переліку вирішуваних завдань, особливостей зберігання, опрацювання та передачі інформації дозволив виділити можливі цілі порушника.

Запропоновані методи та алгоритми є універсальними і можуть бути застосовані для вирішення широкого кола оптимізаційних завдань.

5.3. Оптимізації складу комплексів засобів захисту інформаційно-комунікаційного середовища

На підставі запропонованих у розділах посібника 2, 3, 4 моделей, можна формалізувати завдання оптимізації складу комплексів СЗІ ІКС та АСК за такими критеріями:

1) мінімум ймовірності досягнення порушником усіх цілей;

2) мінімум середнього рівня втрат ІКС та АСК від реалізації порушником усіх цілей;

3) максимум ймовірності успішної протидії СЗІ реалізації всіх цілей порушником;

4) мінімум значення інтегрального показника «вартість-ризик».

У результаті проведеного аналізу доцільним є оптимізувати склад комплексу СЗІ для ІКС та АСК за такими критеріями:

1. За критерієм мінімуму ймовірності досягнення порушником усіх своїх цілей: визначити такі значення для модулів СЗІ x_{jm} ($j \in B_{p_a}$, $j \neq 0$; $m \in N_j^{p_a}$; $p_a=1,2,\ldots, PA$), що

$$P^{p} = P^{p}(X) = \min_{X} \prod_{p_a=1}^{PA} \sum_{\substack{j \in G^{p_a} \\ I^{p_a-1}}} P_j^{p_a} p_{jn_{p_a}}, \qquad (5.37)$$

при обмеженнях: $C^{з} \le C^{з}_{доп}$,

$$\sum_{\substack{j \in G^{p_a} \\ I^{p_a-1}}} P_j^{p_a} p_{jn_{p_a}} \le P^{p}_{p_a доп}, \; p_a = 1,2,..., PA,$$

$x_{jm} = \{0,1\}, (j \in B_{p_a}, j \ne 0; m \in N_j^{p_a}; \; p_a = 1,2,..., PA)$,

де $C^{з}_{доп}$ – максимально припустиме значення вартості СЗІ;

m – засіб СЗІ на *j-му* рубежі (наприклад, на вузлі ІКС - *ит**.).

2. За критерієм мінімуму середнього рівня втрат ІКС та АСК від дій порушника: визначити такі значення x_{jm} ($j \in B_{p_a}, j \ne 0$; $m \in N_j^{p_a}$; p_a=1,2,..., PA), що

$$C^{p} = C^{p}(X) = \min_{X} \sum_{p_a=1}^{PA} \sum_{\substack{j \in B_{p_a}, \\ j \ne 0}} P_j^{p_a} c_{j_{p_a}}, \qquad (5.38)$$

де $c_{j_{p_a}}$ – обсяг втрат ІКС та АСК від порушення конфіденційності інформації, обсяг втрат від невиконання ряду робіт, вартість відновлення системи захисту з реалізації порушником p_a -*ї* загрози,

при обмеженнях: $C^{з} \le C^{з}_{доп}$,

$$\sum_{\substack{j \in G^{p_a} \\ I^{p_a-1}}} P_j^{p_a} p_{jn_{p_a}} \le P^{p}_{p_a доп}, \; p_a = 1,2,..., PA,$$

$x_{jm} = \{0,1\}, (j \in B_{p_a}, j \ne 0; \; m \in N_j^{p_a}; \; p_a = 1,2,..., PA)$.

3. За критерієм максимуму ймовірності успішної протидії СЗІ ІКС та АСК діям порушника: визначити такі значення x_{jm} ($j \in B_{p_a}, j \neq 0, m \in N_j^{p_a}; \quad p_a = 1,2,...,PA$), що

$$P^з = P^з(X) = \max_X \prod_{p_a=1}^{PA} \left(1 - \sum_{j \in G^{p_a}_{I^{p_a}-1}} P_j^{p_a} p_{jn_{p_a}} \right), \tag{5.39}$$

при тих же обмеженнях.

4. За критерієм мінімуму інтегрального показника «вартість-ризик»: визначити значення x_{jm} ($j \in B_{p_a}, j \neq 0$; $m \in N_j^{p_a}$; $p_a = 1,2,...,PA$),

$$S_{c-p} = S_{c-p}(X) = \min_X \left(\begin{array}{l} \sum\limits_{p_a=1}^{PA} \sum\limits_{j \in B_{p_a}} \sum\limits_{m \in N_j^{p_a}} c^з_{jm} x_{jm} \\ + \sum\limits_{\substack{p_a=1 \\ j \neq 0}}^{PA} \sum\limits_{j \in B_{p_a},} P_j^{p_a} c_{jp_a} \end{array} \right), \tag{5.40}$$

де $S_{c\text{-}p}$ – значення інтегрального показника «вартість-ризик»;

$P^p_{p_a доп}$ – припустиме значення ймовірності реалізації порушником p_a - *ї* мети, у разі НСД до ІКС

при таких обмеженнях

$$\sum_{j \in G^{p_a}_{I^{p_a}-1}} P_j^{p_a} p_{jn_{p_a}} \leq P^p_{p_a доп}, \; p_a = 1,2,..., PA,$$

$$x_{jm} = \{0,1\}, (j \in B_{p_a}, j \neq 0; \; m \in N_j^{p_a}; \; p_a = 1,2,..., PA).$$

Відповідно до цілей, завдання визначення оптимального складу комплексу СЗІ ІКС або АСК пропонується вирішувати за критерієм максимуму ймовірності успішної протидії СЗІ реалізації всіх цілей порушником. Дане завдання належить до класу завдань цілочисельного програмування з булевими змінними. Для його вирішення використовуються методи динамічного програмування.

На підставі накопичених статистичних даних (див. розділ 1) щодо діяльності порушників в аналогічних системах обчислювальних засобів, аналізу можливих шляхів реалізації ними виділених цілей, нами було складено перелік можливих загроз інформації і побудовані графи станів системи, що захищається, при реалізації порушником кожної зі своїх цілей (див. розділ 1, 3). Аналіз способів реалізації виділених загроз (див. розділи 1, 2) дозволив скласти перелік засобів і методів захисту інформації, потенційно придатних для включення в комплекс засобів захисту [4-15 та ін.]. Вирішення завдання оптимізації складу комплексів ЗЗІ здійснювалося для різних значень обмеження на вартість системи захисту ІКС та АСК [10].

На підставі аналізу методів прогнозування результатів інвестиційного проектування, і використовуючи раніше викладену модель оптимізації витрат на комплекс СЗІ, і управління ІБ у АПК, пропонується економетрична модель для системи підтримки прийняття рішення по вибору оптимальної стратегії управління інвестиційним проектуванням у ІБ господарюючого суб'єкта (ГС). Загальна складається з реалізації таких основних етапів [10].

1. Ґрунтуючись на даних системного аналізу, визначаються можливі стратегії розвитку ГС, його ІТ структури і завдань забезпечення ІБ.

2. Для визначення прогнозних обсягів послуг, використовується метод множинного регресійного аналізу. На основі залежності функції (обсяг реалізації) від факторів (собівартість, ціна реалізації, індекс споживчих цін, витрати на рекламу і ін.), будується модель множинної регресії, яка використовується як прогнозна модель у вигляді:

$$UP = A + a_1 \times x_1 + a_2 \times x_2 + a_3 \times x_3 + \ldots + a_n \times x_n , \qquad (5.41)$$

де UP – прогнозний обсяг послуг;

x_j , $j \in \overline{1,m}$ – незалежні змінні;

A , a_j – відповідно, константа та коефіцієнти рівняння регресії.

3. Формуються прогнозні значення обсягів реалізації послуг на наступний період часу для платіжної матриці шляхом варіювання значень змінних у

відповідності з безліччю пропонованих стратегій $C = \{C_k\}$ (зокрема стратегій, що передбачають розвиток ІТ структури компанії і відповідних ЗЗІ), $k \in \overline{1,q}$), де q – кількість стратегій, і значеннями можливих станів ринкової кон'юнктури X_j, $j \in \overline{1,m}$, де m – кількість станів ринкової кон'юнктури; формується матриця обсягів реалізації послуг – $UP = \|up_{kj}\|, k \in \overline{1,q}, j \in \overline{1,m}.$

В якості *k -оїй* допустимої стратегії управління ІБ C_k ($k \in \overline{1,q}$) пропонується розглядати сукупність дій ГС, які характеризуються рівнем витрат на ЗЗІ, певною ціновою політикою, бюджетом і іншими внутрішніми чинниками.

В якості стану ринкової кон'юнктури розглядаються різні поєднання зовнішніх, незалежних від ГС чинників (ємкість ринку, інфляція, питання ІБ і так далі), тобто X_j – це *j -е* прогнозне значення стану ринку перевезень, що характеризується ємкістю ринку ТП, певним рівнем інфляції та іншими, незалежними від господарюючого суб'єкта, зовнішніми чинниками.

4. За значеннями даних матриці, UP, для стратегій ІБ, що реалізуються, визначаються оцінки за максимінним критерієм, які забезпечують гарантовано найбільшу перевагу в найгірших умовах:

$$W = \max_{k \in \overline{1,q}} \min_{j \in \overline{1,m}} up_{kj}.$$

5. Формується таблиця ризиків, у тому числі з питань ІБ, на перетині рядків і стовпців якої розміщуються значення величини ризику при реалізації даної стратегії при конкретному стані ринкової кон'юнктури, які розраховуються за формулою:

$$R_{kj} = \max_{k \in \overline{1,q}} up_{kj} - up_{kj}, \; k \in \overline{1,q}, \; j \in \overline{1,m}, \qquad (5.42)$$

де $\max_{k \in \overline{1,q}} up_{kj}$ – максимально можливий обсяг продажів ТП при фіксованому *j -ому* стану ринку X_j;

up_{kj} – обсяг продажів при реалізації фіксованої *k -оїй* стратегії C_k ($k \in \overline{1,q}$) і фіксованому стані ринкової кон'юнктури та ІБ X_j ($j \in \overline{1,m}$).

6. Значення даних матриці *UP*, сформованій в п.3, використовуються для обчислення мінімаксних оцінок стратегій (за Севіджем), що визначають гарантоване найменше значення ризику в як найгіршій ситуації: $S_{k_s} = \min\limits_{k\in 1,q} \max\limits_{j\in \overline{1.m}} R_{kj}$.

7. Для знаходження компромісного положення між песимістичною оцінкою за критерієм Вальда (*W*) та оптимістичною мінімаксною оцінкою (*S*), визначається значення по критерію Гурвіца (*G*) за формулою:

$G = \max\limits_{k\in 1,q}\left(\beta \times \min\limits_{j\in 1,m} up_{kj} + (1-\beta)\times \max\limits_{j\in 1,m} up_{kj}\right)$, де β – фіксований показник песимізму-оптимізму, який визначається експертним шляхом на основі аналізу конкурентних переваг ГС і такий, що $\beta \in [0;1]$.

8. Після оцінки різних варіантів декількома критеріями, приймається рішення: якщо рекомендації співпадають, найкраще рішення обирається з більшою впевненістю; якщо спостерігається протиріччя рекомендацій, то остаточне рішення приймається з урахуванням його переваг і недоліків; наприклад, вибирається та стратегія забезпечення ІБ, яка виявилася оптимальною хоча б для двох критеріїв; якщо отримані різні стратегії для всіх трьох критеріїв, треба варіювати значеннями показника песимізму-оптимізму в критерії Гурвіца або змінити дані, наприклад, в можливих станах ринкової кон'юнктури. З урахуванням вищевикладеного, авторами пропонується наступний формалізований алгоритм побудови економетричної моделі з метою вибору оптимальної стратегії управління інвестиційним проектуванням СІБ для господарюючого суб'єкта АПК.

Крок 0. Формування початкових даних. Формується матриця параметрів $X = \|x_j\| \;\; j \in \overline{1,m}$, де m – кількість параметрів (витрат);

формується безліч припустимих стратегій $C = \{C_k\}$, $k \in \overline{1,q}$, де q – кількість стратегій

Крок 1. Вироджений крок алгоритму. За наслідками множинної регресії обчислюються прогнозні значення об'єму реалізації продукції на наступний

період часу і формується матриця $UP = \left\| up_{kj} \right\|, up_{kj} = A + a_{kj} \times x_{kj}$.

Крок 2. Загальний крок алгоритму. Параметр *k:=1*.

2.1. Для k-*оїї* стратегії C_k визначається значення критерію Вальда: $W_k = \min\limits_j up_{kj}$, $k \in \overline{1,q}$; значення критерію Севіджа: $S_k = \max\limits_{k \in \overline{1,q}} up_{kj} - up_{kj}$;

значення критерію Гурвіца: $G = \max\limits_{k \in \overline{1,q}} \left(\beta \times \min\limits_{j \in \overline{1,m}} up_{kj} + (1-\beta) \times \max\limits_{j \in \overline{1,m}} u_{pkj} \right)$, де фіксоване значення $\beta \in [0;1]$.

2.2. Для кожної *k-оїї* стратегії C_k, аналогічно кроку 2.1, обчислюються значення критеріїв Вальда, Севіджа і Гурвіца і формується вектор значень результатів (W_k, S_k, G_k).

Крок 3. Для $k < q$, *k:=k+1* – перехід на

Крок 3.1. Якщо $k \geq q$, то перехід до кроку N.

Крок N. Фінальний крок алгоритму. Обчислюються:

для критерію Вальда значення $S_{k_W} = \max\limits_{k \in \overline{1,q}} W_k$;

для критерію Севіджа значення $S_{k_s} = \min\limits_{k \in \overline{1,q}} \max\limits_{j \in \overline{1.m}} S_k$;

для критерію Гурвіца значення $S_{k_G} = \max\limits_{k \in \overline{1,q}} G_k$.

Формується оптимальна стратегія розвитку СІБ із умов:

$$\begin{cases} \text{якщо } S_{k_W} = S_{k_S} = S_{k_G}, \text{то } S_{k_e} = S_{k_W}; \\ \quad \begin{cases} \text{якщо } \left(S_{k_W} = S_{k_G}\right) \vee \left(S_{k_W} = S_{k_S}\right) \vee \left(S_{k_S} \neq S_{k_G}\right), \text{то } S_{k_e} = S_{k_W}; \\ \text{якщо } \left(S_{k_W} = S_{k_G}\right) \vee \left(S_{k_W} \neq S_{k_S}\right) \vee \left(S_{k_S} = S_{k_G}\right), \text{то } S_{k_e} = S_{k_G}; \\ \text{якщо } \left(S_{k_W} \neq S_{k_G}\right) \vee \left(S_{k_W} = S_{k_S}\right) \vee \left(S_{k_S} = S_{k_G}\right), \text{то } S_{k_e} = S_{k_S}; \end{cases} \\ \text{якщо } S_{k_W} \neq S_{k_S} \neq S_{k_G}, \text{то перехід на крок 0 та зміна початкових даних.} \end{cases} \quad (5.43)$$

Даний алгоритм був у відповідному модулі програми «Аналізатор загроз» [10].

На рис. 5.4 наведено загальні результати моделювання, з яких видно, що зі збільшенням обсягу асигнувань на ЗЗІ ІКС та АСК ймовірність реалізації порушником усіх цілей значно знижується.

На рис. 5.5 представлена залежність інтегрального показника загальних витрат системи передачі та опрацювання інформації, пов'язаних із втратами від дій порушника і витратами на організацію ЗЗІ, від ймовірності успішної протидії системи захисту його діям. Як бачимо, дана залежність має явно виражений мінімум, у якому значення витрат на організацію системи захисту та втрат від дій порушника рівні. Це свідчить про те, що починаючи з цієї точки рівень витрат на систему захисту починає перевищувати рівень втрат від дій порушника і тому основну частку в значенні інтегрального показника становить сукупна вартість засобів захисту. Отже, можна зробити висновок, що раціональний обсяг витрат на організацію ЗЗІ лежить правіше від точки мінімуму даної залежності.

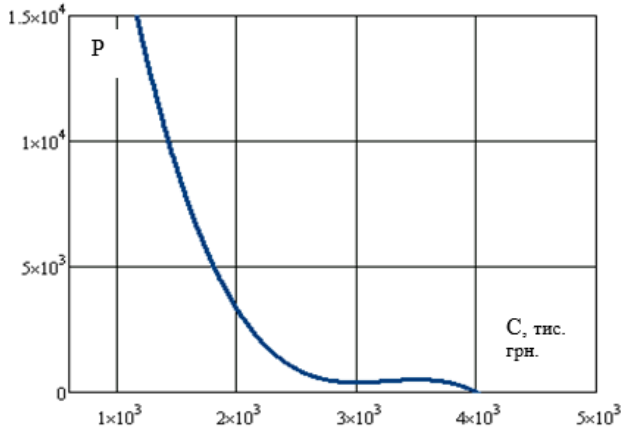


Рис. 5.4. Залежність ймовірності реалізації всіх цілей зловмисником від вартості комплексів СЗІ ІКС

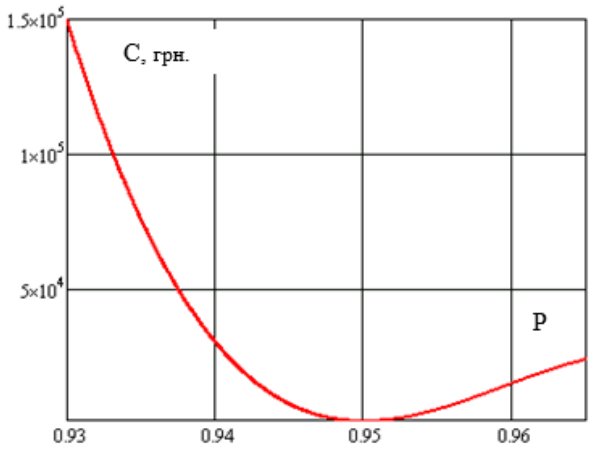


Рис. 5.5. Інтегральний показник загальних витрат на ІКС від ймовірності успішної протидії ЗЗІ дій зловмисника

Так, при обсязі витрат на організацію СЗІ по критичних вузлах ІКС порядку 5254 у.о. ймовірність досягнення порушником всіх своїх цілей становить 10^{-2}. Причому, дана залежність носить явно виражений експонентний з негативним коефіцієнтом характер. Крім того, з рис. 5.4, 5.5, видно, що збільшення асигнувань на організацію системи захисту інформації вище за

певний рівень (вище 13135 у.о.) недоцільне, оскільки не призводить до значного підвищення ефективності системи захисту.

Загальна схема процесу проектування та оптимізації складу СЗІ ІКС представлена на рис. 5.6.

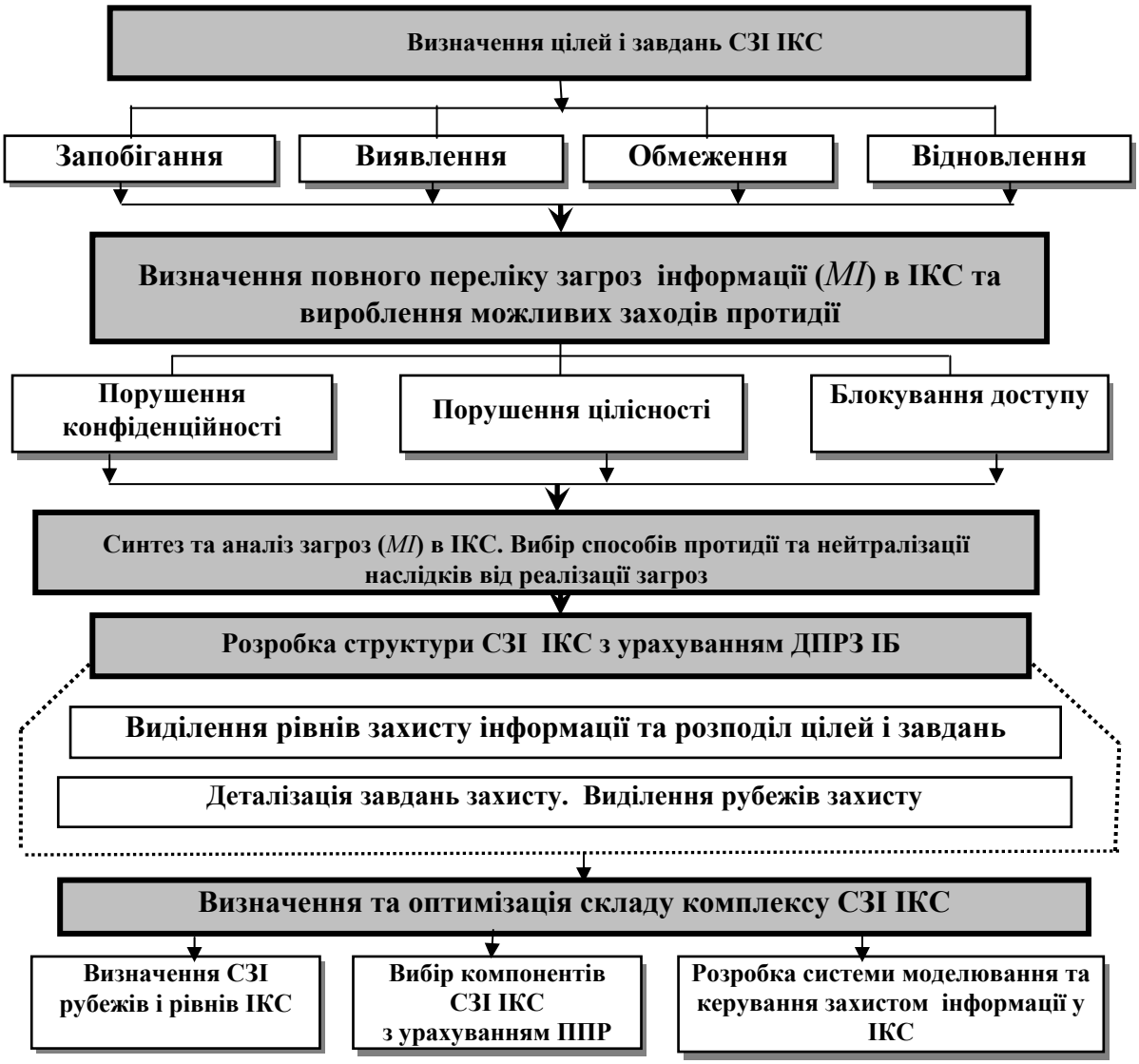


Рис. 5.6. Загальна схема процесу проектування та оптимізації складу СЗІ ІКС

Ураховуючи все вищесказане, та використовуючи моделі, представлені у розділах 2, 3 та 4, пропонуємо загальну методику управління інформаційно-обчислювальним процесом для забезпечення ІБ ІКС, яка представлена на рис. 5.7.

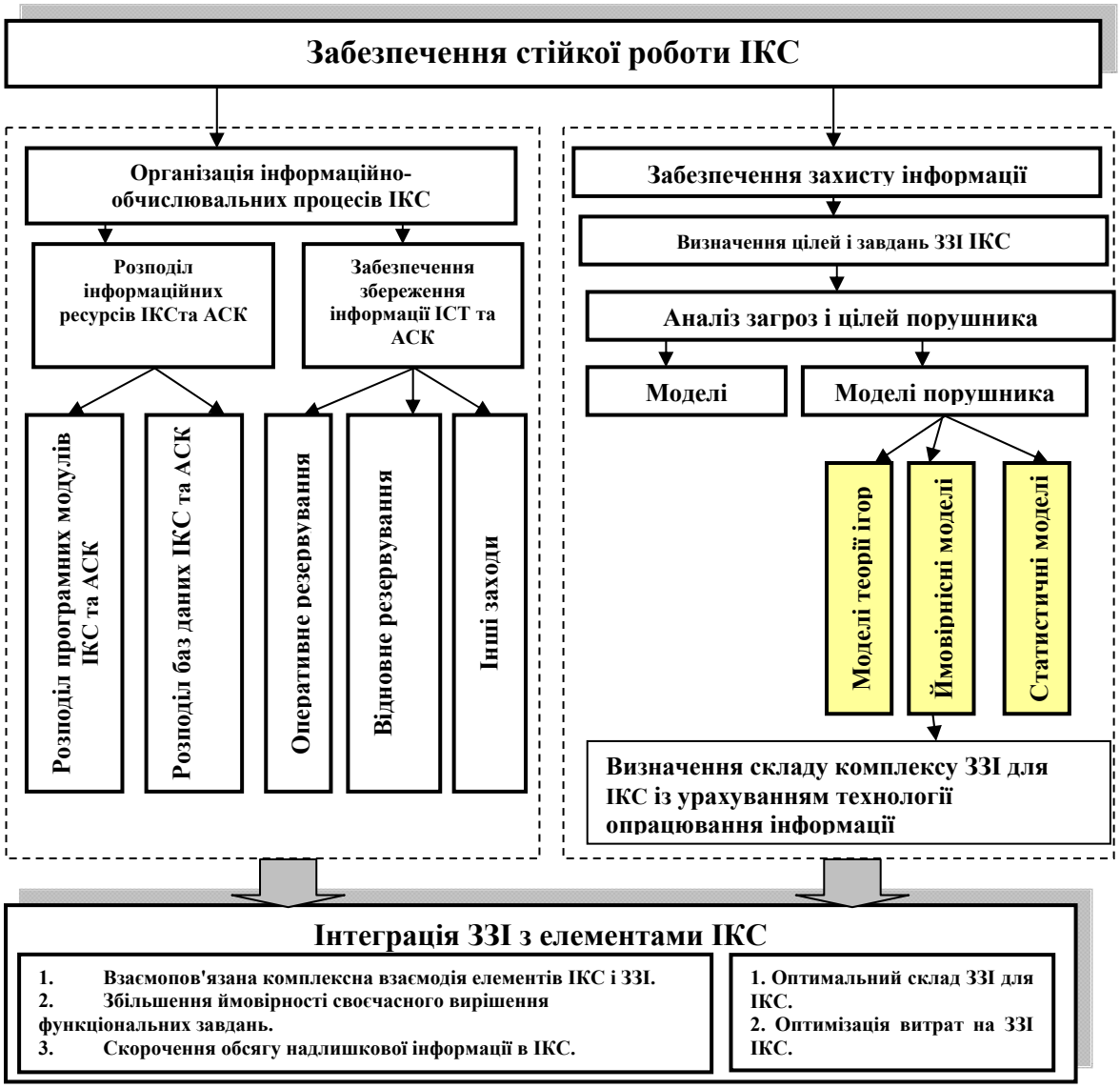


Рис. 5.7. Методика процесу забезпечення стійкості інформаційно-обчислювального процесу, схоронності та захищеності ІКС

5.4. Висновки до розділу 5

В результаті проведених у даному розділі роботи досліджень можна зробити наступні висновки.

1. Встановлено, що при розробці сучасних ЗЗІ ІКС доводиться враховувати велику кількість чинників, які впливають на ефективність їх функціонування, що ускладнює знаходження аналітичних оцінок з вибору узагальненого критерію оптимальності їх структури.

2. Запропоновано нові підходи до вирішення завдання оптимізації складу обраної структури ЗЗІ ІКС за різними критеріями.

3. Формалізовано завдання оптимізації складу комплексів ЗЗІ ІКС за такими критеріями - мінімум ймовірності досягнення порушником усіх цілей; мінімум середнього рівня втрат ІКС та АСК від реалізації порушником усіх цілей; максимум ймовірності успішної протидії ЗЗІ реалізації всіх цілей порушником; мінімум значення інтегрального показника «вартість-ризик».

4. Розроблена економетрична модель для системи підтримки прийняття рішення по вибору оптимальної стратегії управління інвестиційним проектуванням у систему інформаційної безпеки та захисту інформації господарюючого суб'єкта.

5. Запропоновано вирішувати завдання визначення оптимального складу комплексу ЗЗІ ІКС за критерієм максимуму ймовірності успішної протидії СЗІ реалізації всіх цілей порушником. Запропоновано математичну модель оптимізації структурно-технологічного резерву критично важливого програмного забезпечення ІКС.

6. Розглянуто особливості застосування методів дискретної оптимізації для вирішення завдань забезпечення захисту критичної інформації ІКС. Запропоновано алгоритм оцінки меж рішенням двоїстої задачі з визначенням порядку розгалуження для вирішення завдань забезпечення захисту інформації в ІКС. Аналіз результатів показав, що застосування способу попереднього

визначення порядку розгалуження змінних разом із симплекс-методом оцінки меж рішення дозволяє скоротити час вирішення завдань в 5–20 разів.

7. Отримано результати, які свідчать, що обсяг інформації, яка циркулює в системі, за рахунок раціонального розподілу ПМ та ІМ ІКС та АСК зменшився на 17–20%, при цьому більш ніж на 35% зросла ймовірність вирішення всіх завдань в системі при мінімальному обсязі відновного резерву кожного ПМ та ІМ, при одночасному збільшенні стійкості інформаційно-обчислювального процесу. Ймовірність вирішення всіх завдань у системі з урахуванням відновного резервування при обсязі резерву в одну копію зросла на 28–30% при збільшенні ймовірності вирішення кожного із завдань не нижче ніж на 5–7%.

8. Для подальшого вирішення завдань навчального посібника та з метою перевірки отриманих у розділах 2–5 результатів, потрібно провести експериментальні дослідження з інтелектуального розпізнавання загроз інформаційним системам в умовах реалізації КНІ. Експериментальні дослідження, зокрема, потрібні для тестування розробленої експертної системи «Аналізатор загроз», яка дозволяє виконувати розпізнавання загроз ІБ, збирати інформацію про стан захисту ІС та АСК у мережі підприємства, а також, надає допомогу при ухваленні рішення по вибору оптимальної стратегії управління інвестиційним проектуванням СЗІ для господарюючого суб'єкта.

РОЗДІЛ 6
ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ З ІНТЕЛЕКТУАЛЬНОГО РОЗПІЗНАВАННЯ ЗАГРОЗ ІНФОРМАЦІЙНИМ СИСТЕМАМ В УМОВАХ РЕАЛІЗАЦІЇ КОМП'ЮТЕРНИХ НАПАДІВ НА ІНФОРМАЦІЮ

6.1. Методологія проведення експериментального дослідження

Сьогодні багато суб'єктів господарської діяльності змушені забезпечувати постійну доступність своїх інформаційних ресурсів стороннім особам. Тому під час проектування, впровадження та модернізації інфотелекомунікаційної інфраструктури підприємства постійно піднімаються питання оцінки прийнятих технічних рішень, що стосуються продуктивності та стійкості компонентів мережі, засобів системи забезпечення ІБ при роботі в умовах штатного і підвищеного навантаження, зокрема впливу КНІ.

При розширенні ЛОМ підприємства і включення до неї нових модулів, з'являються проблеми, пов'язані з погіршенням швидкісних характеристик роботи мережі і АС (ІС). З огляду на це, складно визначити, які складові інфраструктури необхідно модернізувати. До цього також додається постійне зростання складності ЗЗІ, що підвищує вимоги до кваліфікації обслуговуючого їх персоналу.

На проблему зниження продуктивності інформаційних систем у результаті навмисного втручання в їх роботу з боку зловмисників, наприклад, при КНІ, стали звертати увагу з 80-х років минулого століття [4-15 та ін.].

Щорічно фіксується тисячі зламів сайтів, серверів, програм, баз даних тощо. Практично щодня в засобах масової інформації з'являються повідомлення про незаконне проникнення в корпоративні мережі великих компаній, у мережі Internet викладаються конфіденційні бази даних міністерств, банків та підприємств і т. д. Особливо актуальною стає ця проблема у випадках, коли зовнішні, загальнодоступні, інформаційні ресурси компанії взаємодіють

безпосередньо з її внутрішніми корпоративними ресурсами. Причому дана проблема актуальна як при моделюванні різних класів атак, так і при розробці алгоритмів керування ризиками при захисті інформаційних ресурсів суб'єктів господарської діяльності та фізичних осіб.

Комплексний підхід до забезпечення безпеки ІС (АС та АСК) компанії включає проектування СЗІ, розробку політики ІБ, а також проведення «тесту на проникнення» (penetration test – PT [4-15]) для перевірки ступеня захищеності інформаційних ресурсів ззовні.

Тести на проникнення допомагають на практиці отримати об'єктивну та незалежну оцінку того, наскільки легко здійснити НСД до ресурсів ЛОМ чи сайту компанії. Тобто, тест на проникнення – це моделювання дій зловмисників із проникнення в інформаційну систему в умовах, максимально наближених до тих, які виникають при атаці зловмисників [4-15].

Основні переваги аналітичних методів моделювання характеристик ЛОМ і ІС (АС) – відносна простота відповідних розрахунків і порівняно малі витрати часу на обчислення. Але більшість аналітичних моделей дозволяють отримувати тільки середні значення шуканих параметрів у стаціонарному режимі функціонування ЛОМ і ІС (АС).

Необхідність проведення «тесту на проникнення» [4-15] обумовлена, перш за все, потребою надати керівництву компанії достовірну інформацію про реальний стан ІБ. Нерідко менеджери знаходяться в стані ілюзорної захищеності і впевнені, що кошти, вкладені в програмні продукти (антивірусні програми, фаервол), самі по собі роблять ІКС та АСК стійкими до атак зловмисників. Результатами подібних ілюзій можуть стати як прямі фінансові збитки від втрати конфіденційних даних, так і втрата іміджу при, наприклад, розміщенні на зламаному сайті спотвореної інформації. А це, у свою чергу, може призвести до нівелювання напрацьованих конкурентних переваг і інших, негативних для бізнесу наслідків.

Як правило, великі компанії мають розвинену ІТ – інфраструктуру (див. розділ 1). Таким чином, наявність великої кількості об'єктів тестування

визначає вибір методики проведення відповідних досліджень. Це може бути як створення атакуючого, вірусного або легітимного трафіку, так і емуляція діяльності користувачів або одночасне виконання таких дій. Різнитися можуть точки, у яких створюється навантаження (трафік). Наприклад, може знадобитися провести тестування захищеності мережевого сервісу як від атак зсередини локальної мережі підприємства, так і від атак з мережі Інтернет.

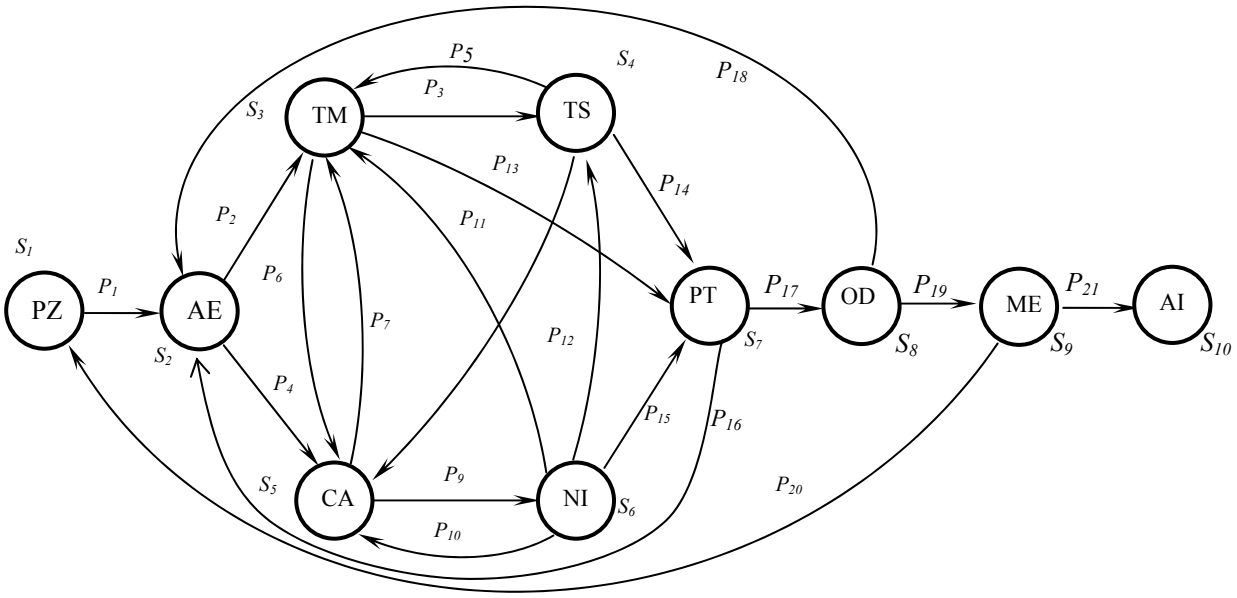
Об'єктами тестів на проникнення, за погодженням з керівництвом, були обрані інформаційно-керуючи системи підприємств ТОВ «Лугавтотранс», що займається наданням транспортних послуг населенню та підприємствам Луганської області (Код за ЄДРПОУ: 30760847), а також ОЦ ДП Придніпровської залізниці. Згідно з методикою проведення тесту на проникнення в ІС підприємств, початковими даними були:

1. Перелік IP-адрес хостів, що утворюють зовнішній периметр ІС.
2. Перелік адрес електронної пошти співробітників, (адреси були отримані в результаті репрезентативної вибірки за співробітниками з різних структурних підрозділів ТОВ «Лугавтотранс», ОЦ ДП Придніпровська залізниця);
3. Дані про встановлені АСК, ІС та АІС підприємств.

Була використана методика, що дозволила найбільш повно змоделювати дії потенційного порушника [4-15], зокрема це: пасивний збір відомостей про ІС Замовника з відкритих джерел; активний збір відомостей про ІС Замовника (підключення до хостів зовнішнього периметра); перевірка можливості проникнення в ІС Замовника за допомогою використання уразливостей мережевих служб, запущених на хостах зовнішнього периметра; перевірка можливості проникнення в ІС Замовника за допомогою троянської програми.

Як потенційні порушники ІБ розглядалася група осіб, які перебувають у змові і в результаті навмисних дій можуть реалізувати різноманітні загрози для ІБ, спрямовані на інформаційні ресурси і завдати моральної та/або матеріальної шкоди інтересам ТОВ «Лугавтотранс», ДЗ, ОЦ ДП Придніпровська залізниця.

Структура формальної моделі процесу організації та проведення експериментальних досліджень ЛОМ і ІС підприємств може бути представлена у вигляді наступного графа (див. рис. 6.1).



Смислове значення вершин графа G : S_1 – постановка завдання експерименту, в ході якого реалізується тест на проникнення (PZ); S_2 – скла-дання алгоритму експериментів для тесту на проникнення в ІС (AE); S_3 – вибір засобів вимірювання трафіку (TM); S_4 – вибір засобів захисту інформації (СЗІ) (TS); S_5 – побудова алгоритму управління ЗЗІ (CA); S_6 – налагодження засобів вимірювання (NI); S_7 – проведення експерименту (тест на проникнення) (PT); S_8 – опрацювання отриманих даних (OD); S_9 – моде-лювання елементів ІС на основі експериментальних даних (OD); S_{10} – аналіз зміни продуктивності ІС в ході тестів на проникнення (AI).

Рис. 6.1. Структура формальної моделі процесу організації та проведення експерименту

В даному графі вершини відповідають станам системи (тобто, зміна станів настає відповідно до дій експериментатора, що виконує роль зловмисника і намагається подолати захист ІС) в процесі організації експериментів, а дуги відповідають зв'язкам між етапами подолання рубежів захисту інформації, і порядок виконання певних дій з боку порушника (порушників): $G = (S, P)$, де $S = (S_1, ..., S_N)$; $P = (P_1, ..., P_M)$.

Відповідно до постановки завдання (PZ), експериментатор будує алгоритм експерименту тесту на проникнення (AE). Даний алгоритм дозволяє намітити дії, необхідні для здійснення поставленого завдання з подолання рубежів захисту ІКС. Також на підставі розробленого алгоритму експерименту проводиться вибір засобів вимірювання мережевого трафіку (TM) в ЛОМ і ІС. На наступному етапі виконується складання алгоритму управління (CA) технічними засобами вимірювання мережевого трафіку в ході тестів на проникнення.

У процесі розробки алгоритму управління і настроювання засобів вимірювання показників продуктивності ІС (NI), можна коригувати склад і функцій окремих технічних засобів (P_7, P_{12}). В результаті отримаємо експериментальні дані, а також ще на першому етапі виявимо можливі помилки в загальному алгоритмі проведення тесту на проникнення. Для усунення помилок виконується корекція (P_{16}). Отримані експериментальні результати піддаються опрацюванню (OD), після якої отримуємо масиви даних для проведення аналізу й подальшого моделювання роботи мережі та інформаційної системи. На цьому етапі проведення досліджень можуть виникати похибки в складанні алгоритму тесту і, відповідно, необхідність його корекції (P_{18}). Наступним етапом досліджень є моделювання процесів передачі запитів і отримання даних (ME) у ІС. У разі незадовільного результату роботи моделі можливе коригування поставленого завдання (P_{20}). Результат дослідження продуктивності ІС (AI) завершує процес експериментальних досліджень тесту на проникнення.

Моделювання роботи модуля обчислювального комплексу ІС підприємств проводилось за таких умов (див. табл. 6.1).

Таблиця 6.1

Умови експериментального моделювання зниження продуктивності ІС підприємств при КНІ

Вели-чина	Значення	Розмір-ність	Примітка
NO	1–6	од.	Кількість каналів обслуговування в модельованій СМО (ІС)
K	1–6	од.	Кількість зайнятих каналів обслуговування
SK	1	од.	Передбачувана кількість запитів, які очікують обслуговування в буфері модуля ВК ІС
QS	1 – 7	од.	Кількість спроб доступу до середовища передачі даних з урахуванням генерованих нападником (i)
UM	6–10	од.	Кількість обчислювальних модулів у мережі
λ (k_1, k_3)	17 – 6000	запит/с	Інтенсивність потоку запитів, що надходять у модуль ВК ІС (сервер)
$X_н$	10	Мбіт/с	Номінальна пропускна здатність середовища передачі даних
τ_{mki}	96	біт	Міжкадровий інтервал
$\tau_{оч}$	0,001–0,01	с	Час очікування на обслуговування

Методика проведення вимірювальних експериментів під час тесту на проникнення включає: елементи вимірювального експерименту; етапи проведення тесту на проникнення; отримання характеристик структурованої кабельної системи обчислювальної мережі ІС підприємств; первинне дослідження мережі (одержання списку імен ЕОМ, доступних портів та ін.); виявлення недоліків спеціального ПЗ та архітектури мережі підприємств; підключення аналізатора протоколів до досліджуваного обчислювального комплексу ЛОМ і ІС, АСК; інтерпретація результатів вимірювань.

У ході тесту також розглядалося завдання отримання та управління правами доступу до інформаційних ресурсів ТОВ «Лугавтотранс», ОЦ ДП Придніпровська залізниця.

Змістовна постановка задачі має такий вигляд:

Вектор $V_i = \{v_i\}$ ознак об'єктів доступу (ОД) до ІС (АС):

$$v_i = \begin{cases} 1, \textit{ якщо до вузла } v_i \textit{ дозволено загальний доступ}; \\ 0, \textit{ у протилежному випадку}. \end{cases} \quad (6.1)$$

Матриця $M_{СД} = [\mathrm{m}_{\mathrm{СД\,j,k}}]$ розподілу суб'єктів доступу (СД) до локальних вузлів (ЛВ) ІС (АС):

$$\mathrm{m}_{\mathrm{СД_{j,k}}} = \begin{cases} 1, \textit{ якщо } sd_j \in um_k; \\ 0, \textit{ у протилежному випадку}. \end{cases} \quad (6.2)$$

Матриця $M_{ОД} = [\mathrm{m}_{\mathrm{ОД\,j,k}}]$ розподілу ОД до локальних вузлів (ЛВ) ІС (АС):

$$\mathrm{m}_{\mathrm{ОД_{j,k}}} = \begin{cases} 1, \textit{ якщо } rd_j \in um_k; \\ 0, \textit{ у протилежному випадку}. \end{cases} \quad (6.3)$$

Критерії безпеки ІС ТП $Z_0 = \{z_{0_{i,j}}\}$:

$$z_{0_{i,j}} = \begin{cases} 1, \textit{ якщо } rd_{\mathrm{i}} \textit{ розміщено на вузлі } \mathrm{um_k}; \\ 0, \textit{ у протилежному випадку}. \end{cases} \quad (6.4)$$

де $RD = \{\mathrm{rd_i}\}; \mathrm{i} = \overline{1,\mathrm{I}}$ – множина об'єктів доступу; $SD = \{\mathrm{sd_j}\}; \mathrm{j} = \overline{1,\mathrm{J}}$ – множина суб'єктів доступу; $UM = \{\mathrm{um_k}\}; k = \overline{1,\mathrm{K}}$ – множина локальних вузлів.

Зведена характеристика ймовірних порушників наведена в таблиці 6.2.

Таблиця 6.2

Характеристики ймовірних порушників

Класифікація	Характеристика
За мотивом порушення	Порушення цілісності, конфіденційності, доступності з корисливою чи іншою метою.
За рівнем інформованості та кваліфікації порушника	Порушник має: 1) високий рівень знань; 2) достатні знання для збору інформації, застосування відомих експлойтів та написання власного ПЗ для здійснення КНІ; 3) порушник (и) не є авторизованим користувачем ІС (АС).
За місцем дії	Без безпосереднього (фізичного) доступу на територію об'єкта (зовнішній порушник). Порушник діє віддалено, через мережу Інтернет

При виконанні експериментальних досліджень у межах тесту на проникнення було зроблено такі припущення:

1) обчислювальна мережа підприємства буде здійснювати передачу запитів Ethernet в умовах конкурентного доступу модулів до поділюваного середовища передачі даних, при цьому вплив колізій на роботу обчислювальної мережі не виключається;

2) передбачуване інформаційне навантаження – $\lambda \approx 350$ запитів/с.

На підготовчому етапі робіт з виконання тесту на проникнення було складено опис сервісів, що надаються ІС компанії співробітникам підприємства і користувачам глобальної мережі Інтернет, наприклад, перегляд розкладу руху, попереднє оформлення замовлення на перевезення вантажу та ін. Така попередня оцінка дозволяє виділити основні напрямки, що підлягають аналізу в першу чергу.

Було зроблено припущення, що типові сервіси даних ІС – це сайт компанії, електронна пошта, системи доступу для віддалених клієнтів.

Дослідження продуктивності ЛОМ і ІС підприємств виконувалося за допомогою підключення робочої станції аналізатора протоколів Network Instruments Observer 9.1 до мережі підприємства. Також використовувалися програми програм TRACEROUTE і TRACEPATH.

Для виявлення в АСК пристроїв, що взаємодіють по протоколу Modbus використовувалася утиліта PLCScan.

Проведення тесту на проникнення - складне організаційно-технічне завдання, і один інструмент не може вирішити всі потенційно можливі проблеми. Для вирішення деяких питань у ході досліджень була розроблена програма «Аналізатор загроз» (див. рис. 6.2 – 6.6), призначена для: розпізнавання загроз ІБ ІКС; збору інформації про стан комп'ютерів у мережі підприємств АПК; сканування запущених програм і процесів на АРМ підприємств АПК; визначення рівнів безпеки комп'ютера (АРМ) і можливість подальшого поширення «вірусу» з даного АРМ; фіксації часу «зараження» комп'ютерів у мережі; моделювання процесу ухвалення рішення по вибору оптимальної стратегії управління інвестиційним проектуванням СЗІ для господарюючого суб'єкта АПК; оцінки поточних ризиків НСД до ІС підприємства АПК.

В основу програми покладено припущення про те, що елементи множини функцій безпеки можуть не повністю забезпечувати виконання вимог ІБ на підприємстві, а отже, призводити до зростання показника поточних інформаційного ризиків (див. розділ 2). Задається рівень поточного інформаційного ризику, який вважається прийнятним і не вимагає вживання дорогих заходів протидії спробам НСД.

Програма включає в себе кілька модулів, що можуть функціонувати і як єдиний комплекс, і у вигляді самостійних програмних продуктів.

Модуль 1 – Підпрограма «Ранг», для ранжирування оцінок рівнів ІБ на тестованому підприємстві (більш детально програма ранг описана в роботах [4-15]);

Модулі 2 і 4 – Підпрограми «Матриця моделювання загроз (The interaction matrix classes of threats)», для визначення значущості загроз і уразливостей у ЛОМ і ІС підприємства (більш детально програма The interaction matrix classes of threats описана в роботах [4-15]), див. рис. 6.2;

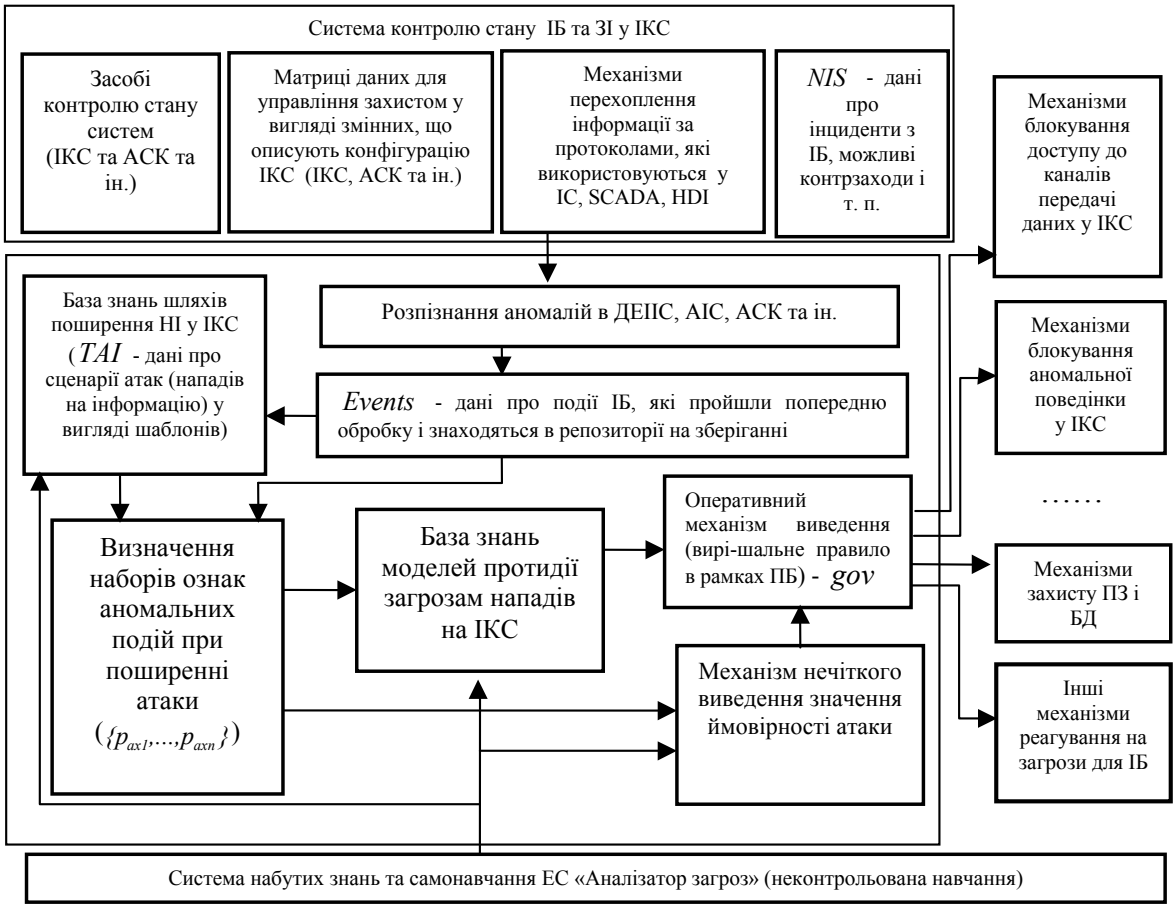


Рис. 6.2. Структура модулів 2 та 4 системи інтелектуального розпізнавання загроз (неконтрольоване навчання)

Модуль 3 – головний модуль програми «Аналізатор загроз».

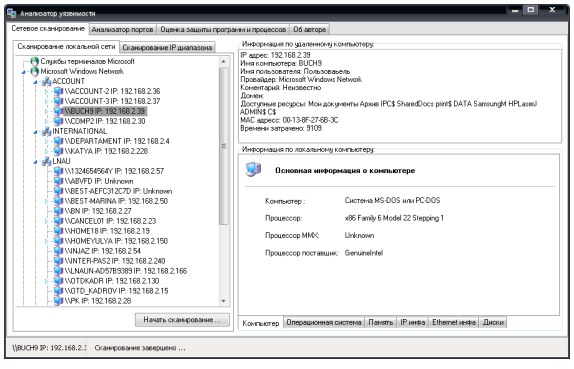


Рис. 6.3. Загальний вид програми «Аналізатор загроз»

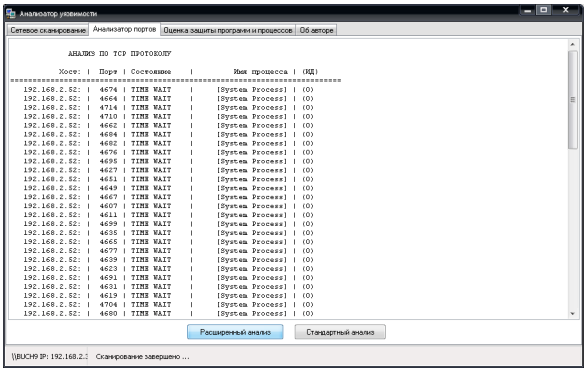


Рис. 6.4. Закладка «Аналізатор портів»

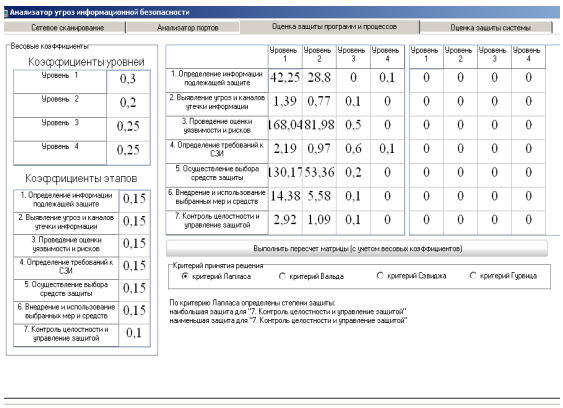


Рис. 6.5. Модуль прийняття рішення по вибору оптимальної стратегії управління інвестиційним проектуванням СЗІ для ГС

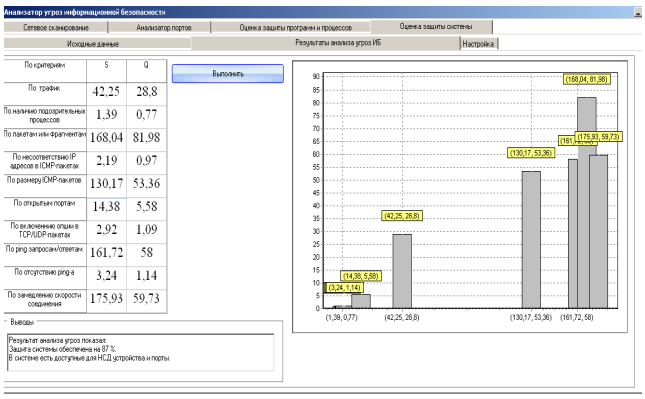


Рис. 6.6. Модуль оцінки ІБ господарюючого суб'єкта

Модуль 4 – система підтримки прийняття рішення по вибору оптимальної стратегії управління інвестиційним проектуванням у ІБ господарюючого суб'єкта.

Схема взаємодії модулів програми показана на рис. 6.7.

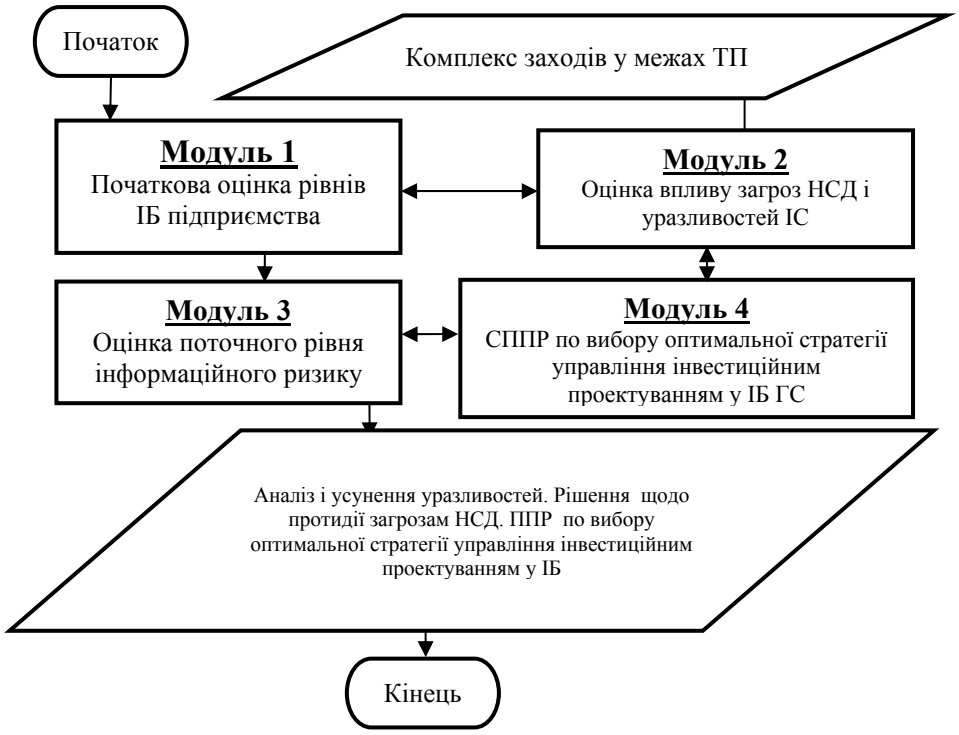


Рис. 6.7. Схема взаємодії модулів програми «Аналізатор загроз»

Основні коди програми «Аналізатор загроз» наведені в додатку В.

На другому етапі модель поповнюється шляхом переведення механізмів захисту ІКС із статусу «потенційні загрози» у статус «активні загрози» і прив'язки активованого механізму до відповідного ешелону СЗІ. Число елементів в підмножині заданих загроз збільшується, як за рахунок включення елементів з множини відомих загроз, так і за рахунок поповнення множини загроз раніше невідомих, див. табл. 2.5, 2.7. Далі можливе розширення множини потенційних механізмів захисту ІКС за рахунок подальшого опису у вигляді нечітких продукційних правил і подальшої реалізації раніше відсутніх механізмів захисту ІКС.

При подальшій адаптації ЕС ІБ ІКС відбувається навчання СЗІ під відсутній механізм захисту інформації, спрямований на нейтралізацію неописаної раніше загрози. Аналіз додаткового нечіткого продукційного правила дозволяє сформувати специфікацію на проектування відсутнього в системі засобу або механізму захисту інформації.

Задаючи порогові значення для величини нечіткого висновку продукційних правил, що визначає ступінь використання *і -го* механізму захисту у формуванні значення підсумкового критерія захищеності системи (див. розділ 2), можна визначати, як не задіяні, так і ефективні механізми в забезпеченні ІБ ІКС. Рішення про розширення класифікацій атак і механізмів захисту проводиться відповідно до системи оцінок достовірності нейтралізації загроз в розрізі окремих механізмів захисту або окремих ешелонів СЗІ та аналогічних оцінок потенційного збитку.

Показник поточних інформаційних ризиків $C_{ППР}$ НДС може приймати різні значення залежно від ряду факторів. Як було зазначено в розділі 2, поточні ризики можуть бути незначними, якщо всі потенційно небезпечні параметри ІС і процесу опрацювання інформації знаходяться у встановлених межах. Або, навпаки, збільшуватися при відхиленні таких параметрів ІБ від норми.

Як цільова функція в програмі «Аналізатор загроз» була прийнята величина очікуваного збитку від НСД, що визначається через міру розбіжності

між реальним та оптимальним механізмами розмежування доступу в компанії (підприємстві):

$$W = \sum_{i=1}^{I} \sum_{j=1}^{J} p_{i,j} \cdot \Delta z_{0_{i,j}}, \tag{6.5}$$

де $P_{i,j} = \{p_{i,j}\}, \mathrm{i} = \overline{1,\mathrm{I}},\ \mathrm{j} = \overline{1,\mathrm{J}}$ – матриця шкоди, зумовленої можливістю НСД до ресурсів; елементи $\{p_{i,j}\}$ визначаються ступенем конфіденційності інформації в ІС і профілем користувача.

6.2. Результати експериментальних досліджень

На першому етапі досліджень було виконано активне сканування всіх портів для кожного з досліджуваних хостів, що дало можливість отримати відомості про ІС ТОВ «Лугавтотранс» (див. рис. 6.8).

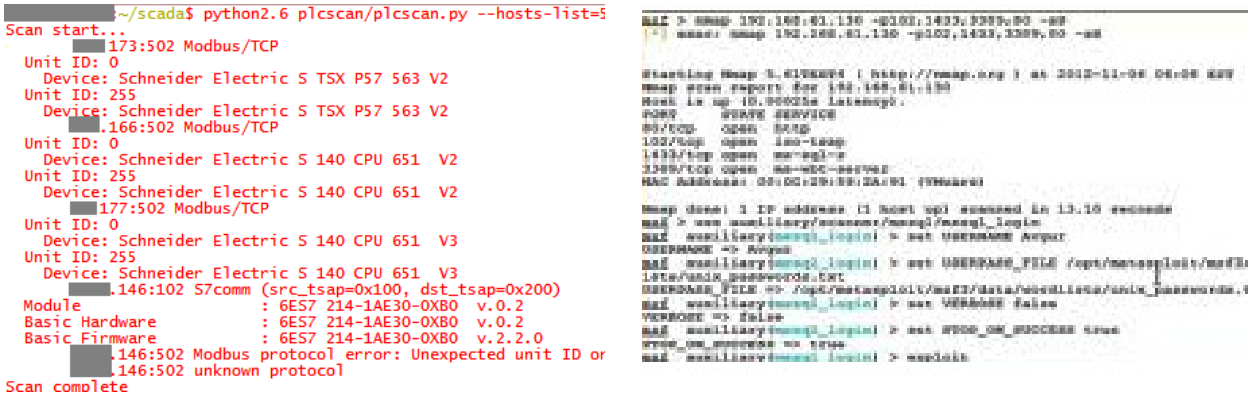
```
BSSID              PWR  Beacons    #Data.  #/s  CH  MB   ENC  CIPHER  AUTH

06:D0:44:B0:88:DA  -35  13         0       0    6   54e. OPN
0A:D0:44:B0:88:DA  -34  12         0       0    6   54e. OPN
00:D0:44:B0:88:DA  -42   9         1       0    6   54e. WPA2 CCMP    PSK
00:26:5A:25:72:AA  -53  10         1       4    6   54e. WPA2 CCMP    PSK
00:1C:DF:88:44:3D  -57   4         0       0    6   54   WPA2 CCMP    PSK
00:22:3F:1B:05:CC  -58  13         0       0    6   54e  WPA2 TKIP    PSK
F0:7D:68:4A:AF:52  -61  10         0       0    6   54e  WPA2 CCMP    PSK
1C:AF:F7:24:5D:7E  -59   9         0       0    6   54e. WPA2 CCMP    PSK
C2:3F:GE:D4:2A:DD  -65   3         0       0    1   54e  WPA2 CCMP    PSK
00:26:5A:30:F4:AC  -62   8         0       0    6   54e. WPA2 CCMP    PSK
00:1B:2F:96:B3:5E  -64   8         1       0    1   54e. WPA2 TKIP    PSK
C0:3F:0E:D4:2A:DC  -65   3         0       0    1   54e  WPA2 CCMP    PSK
00:14:6C:D3:BB:F2  -68   6         0       0   10   54   OPN
00:25:69:51:64:0C  -69   3         0       0    1   54   WPA2 TKIP    PSK
00:D0:44:B0:88:DA  -71   2         0       0    6   54e. WPA2 TKIP    PSK

BSSID              STATION      PWR      Rate      Lost  Packets  Probes
Starting Nmap 5.3SDC1(http://lugavtotrans.com) at 218
Stats: 0:00:33 elapsed: 2 hosts completed(1 up)
Service scan Timing: About 85.71% done: ETC: 22
Nmap scan report for api.home (174.120.83.218)
Host is up(0.047s latency).
PORT     STATE          SERVISE         VERSION

22/tcp   filtered       ssh
88/tcp   open           http?
139/tcp  open           netbios-ssn     Samba smbd 3.x
443/tcp  open           ssl/https?
445/tcp  open           netbios-ssn     Samba smbd 3.x
44567/tcp open          unknown
8080/tcp open           http-proxy?
```

Рис. 6.8. Результати активного сканування

Інформація, отримана за допомогою програм TRACEROUTE і TRACEPATH, дозволила скласти приблизну карту мережі і зробити припущення про встановлені міжмережеві екрани та правила фільтрації мережевого трафіку. Під час активного збору інформації було проведено сканування хостів усього діапазону за допомогою програми NMAP. Для зниження ризику виявлення «порушників» сканування проводилося протягом тривалого часу, з великими інтервалами між скануванням окремих портів та пристроїв у АСК.

Крім того, були досліджені модулі АСК призначені для управління об'єктами системи електропостачання, див. рис. 6.9. Завдяки використанню в АСК уніфікованого протоколу TCP IP система дозволяє передавати будь-яку телеметричну інформацію з різних пристроїв, обладнаних інтерфейсом RS-485 працюючих в протоколі MODBUS, див. рис. 6.9 а).

За допомогою утиліти Metasploit WinCC Harvester отриман доступ до SCADA диспетчера підприємства, див. рис. 6.9 б.



а) б)

Рис. 6.9. Результати активного сканування АСК PLCScan та Metasploit WinCC Harvester

Пасивний збір відомостей про ІС проводився за допомогою загальнодоступних мережевих сервісів – служби DNS, служби WHOIS, а також програм TRACEROUTE, TRACEPATH і т. п. [4-15], web-інтерфейс до яких надають багато сайтів (таким чином, IP-адреса потенційного порушника залишається невідомою для атакованої системи).

Утиліта HOST дозволила отримати перелік всіх доменних імен досліджуваної доменної зони – lugavtotrans.com.

Аналіз доменних імен хостів, що знаходяться в зоні lugavtotrans.com, дозволив зробити припущення про функціональне призначення хостів, які мають записи в базі DNS.

Генерація трафіку, який містить сигнатури, що призводять до несанкціонованого доступу до ресурсів ІС і сервера підприємства, а також генерація пакетів, що містять параметри та характеристики, характерні для відомих атак, здійснювалася за допомогою програми Pandora (ver 1.8.0.502).

Були визначені версії програмного забезпечення мережевих служб на досліджених хостах. Проведене сканування показало коректність вихідних припущень про профіль ІС та АСК підприємств.

Під час проведення тесту на проникнення та для тестування розробленої експертної системи було проведено низку експериментів. В якості вхідних даних для навчання та тестування використовувалася база даних KDD Cup Data [4-15].

Таблиця 6.3 містить результати виявлення атак найбільш поширених КНІ різними методами у порівнянні з ДПРЗ. Як видно з представлених результатів, пропонований метод ДПРЗ показує кращі результати пошуку КНІ. Слід зазначити, що відсоток помилкових спрацювань, коли легітимне з'єднання приймається за атаку, становить менше 10 %. Також, варто відзначити, що в таблиці 6.3 представлені середні результати по 4 класах мережевих атак.

Запропонований підхід, заснований на застосуванні методу ДПРЗ, дозволяє підвищити рівень виявлення мережевих КНІ у ІКС. Виявлення деяких типів атак відбувається з ймовірністю 25–98 % при незначному рівні помилкових спрацювань. Крім цього, запропонований метод не вимогливий до ресурсів ІС і здатний виявляти невідомі типи КНІ.

Таблиця 6.3

Результати ІРЗ мережевих атак під час тесту на проникнення

Методи виявлення атак	Відмова в обслуго-вуванні DoS/ DDoS, %	Скануванні портів ІС з метою отримання конфіден-ційної інформації Probe, %	Отриманням доступу незареєстро-ваного користувача до комп'ютера з боку віддаленої машини R2L, %	Отримання зареєстрованим користувачем права локального супер-користувача (адміністратора) U2R, %
ІРЗ (ДПРЗ)	98,0	95,1	50,5	70,8
Гаусівський класифікатор	82,4	90,2	29,6	22,8
Алгоритм найближчого кластера	97,1	88,8	23,4	22,6
Дерево рішень	97,0	80,8	4,6	10,8

В результаті означеного експерименту для розробленого методу інтелектуального розпізнавання КНІ були отримані наступні результати:

для атак DoS/DDoS – для помилок першого роду (кількість помилкових спрацьовувань) – 10,2%) і помилок другого роду (кількість невиявлених атак) – 2,9%;

для атак Probe – для помилок першого роду – 12,1% і помилок другого роду – 3,1%;

для атак R2L – для помилок першого роду – 9,4% і помилок другого роду – 2,7%;

для атак U2R – для помилок першого роду – 11,3% і помилок другого роду – 3,4%.

Дані результати дозволяють порівняти розроблений метод (ДПРЗ) з дослідженими раніше в роботах [4-15] системами виявлення, див. рис. 6.10.

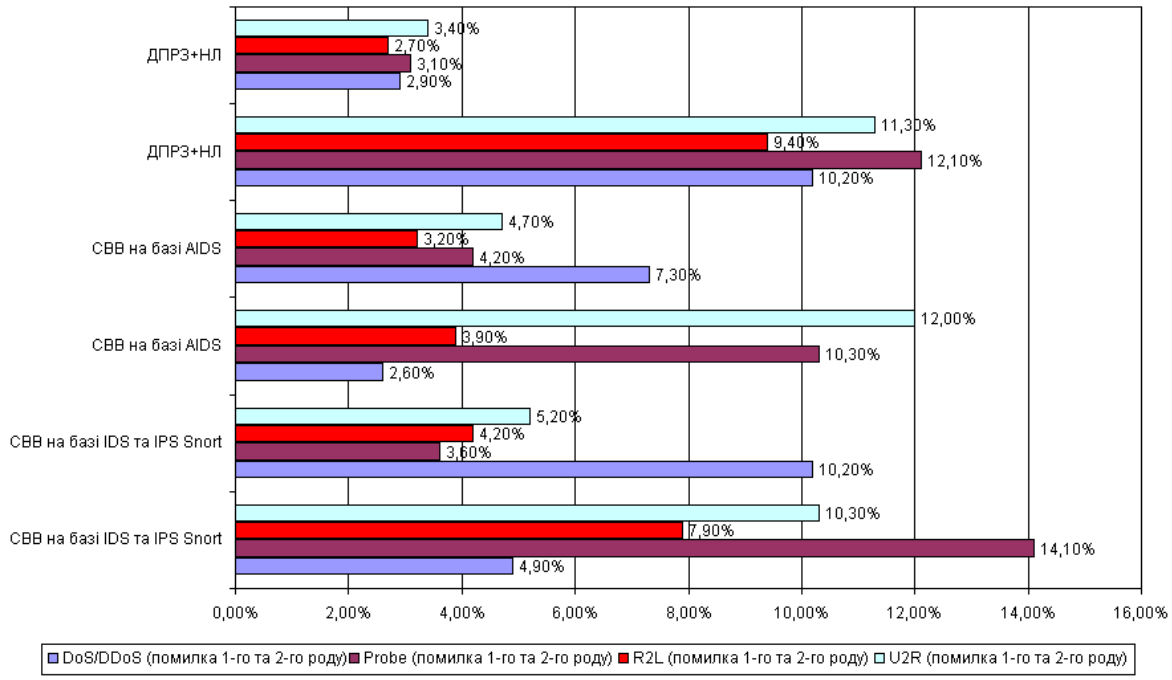
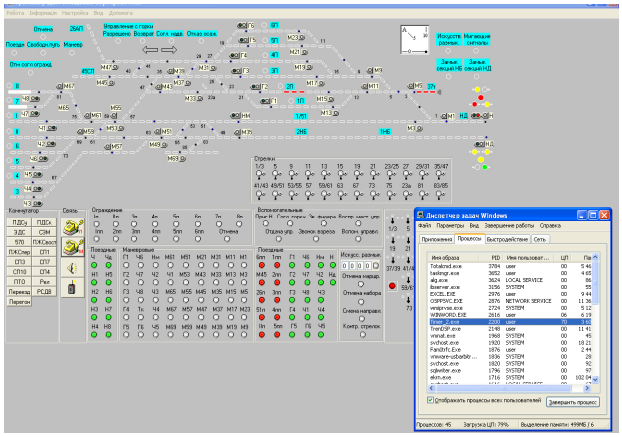
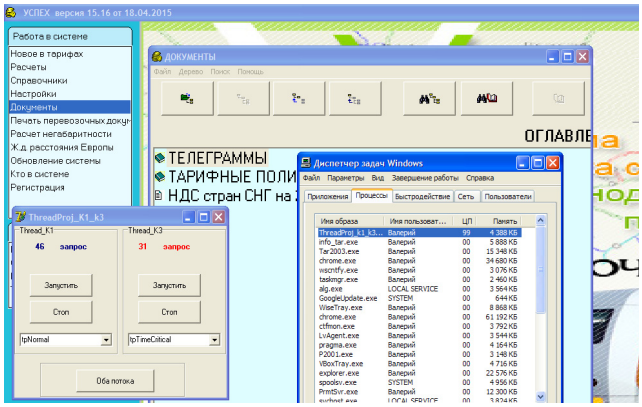


Рис. 6.10. Значення помилок виявлення КНІ першого роду і другого роду для різних систем

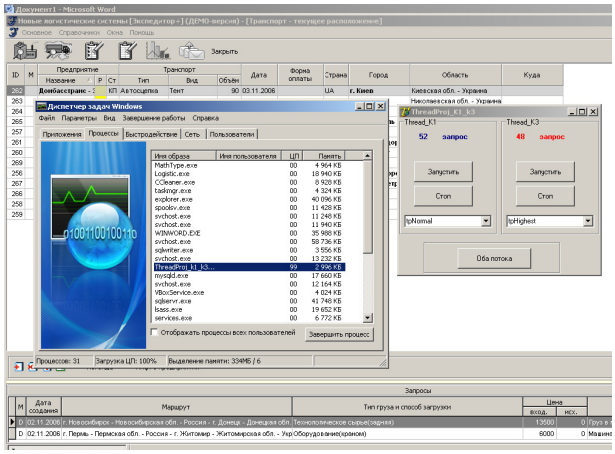
В ході досліджень, з метою створення додаткового навантаження були підготовлені програми – макети вірусів, див. рис. 6.11. Програми-макети вірусів не становили небезпеки для комп'ютерної мережі підприємства, але поширювалися тими ж методами, що й реалізовані в більшості сучасних шкідливих програм.



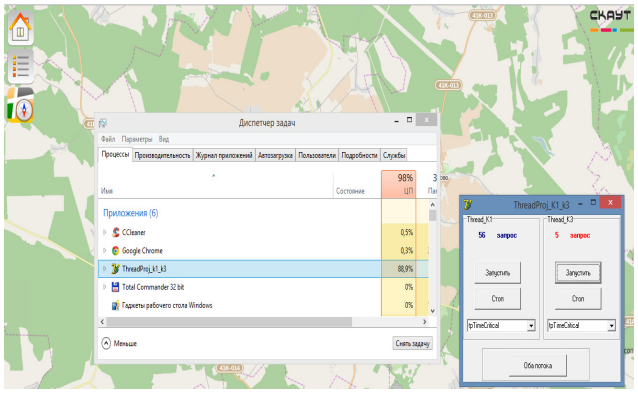
а)



б)



б)



в)

а) вплив макету вірусу на ІС диспетчера: $10 < k_1 < 100$ й $1000 < k_3 < 100000$; Завантаження АРМ: 70 % – 60 с;

б) вплив макету вірусу на ІС ОЦ ДП Придніпровська залізниці «Планування перевезень» та «Документообіг» (потік Бартлетта): $25 < k_1 < 200$ й $1000 < k_3 < 100000$; Завантаження АРМ: 70 % – 60 с;

в) вплив макету вірусу на ІС «Експедитор» ТОВ «Лугавтотранс» (потік Бартлетта) $100 < k_1 < 500$ й $1000 < k_3 < 100000$; Завантаження ПК: 99 % – 30 с;

г) вплив макету вірусу на ІС супутникової навігації диспетчера $100 < k_1 < 500$ й $1000 < k_3 < 100000$; Завантаження ІС: 89 % – 30 с;

Рис. 6.11. Результати впливу макетів вірусів на ІС підприємств АПК

При потраплянні програми – «вірусу» одним із перерахованих способів на АРМ і його подальшому запуску програма проводить такі дії:

1) знижується оцінка локальної антивірусної безпеки (тому що «вірус» уже був запущений);

2) проводиться пошук серед запущених процесів антивірусних програм. Якщо таких процесів немає, то оцінка локальної антивірусної безпеки знижується на кілька балів;

3) виконуються спроби доступу до розділів автозапуску і спеціалізованих папок користувача (наприклад, папка «Автозавантаження»). При успішному запису «вірусу» в автозапуск оцінка знижується;

4) проводиться спроба завантаження тестового файлу з локальної мережі. Успішне завантаження тестового файлу свідчить про відсутність або неправильне використання мережевого екрану, оцінка знижується.

На основі експериментальних даних було отримано такий вираз, що характеризує залежність кількості «заражених» комп'ютерів від часу для розрахованих середніх оцінок безпеки комп'ютерної мережі підприємства:

$$x_{n+1} = 1 - (1 - P_{ср}) \cdot (1 - x_n)_n^{P_{ср} \cdot x}, \tag{6.6}$$

де $P_{ср}$ – середня ймовірність зараження комп'ютерів, що обчислюється виходячи з різниць (*1 – M/10*), де *M* – оцінка локальної антивірусної безпеки комп'ютера за десятибальною шкалою, отримувана за допомогою аналізу ряду чинників: наявності встановлених антивірусних програм у запущених процесах, наявності і якості роботи мережевого екрану, активності користувача в годинах роботи за комп'ютером, кількості спільних ресурсів і прав доступу до них, можливості автозапуску програм із зовнішніх носіїв інформації та частота їх використання, права доступу до параметрів автозавантаження системного реєстру.

Дані «віруси» представляють собою клієнтські програми, які, «заразивши» АРМ, збирали інформацію про рівень безпеки кінцевого ПК, наявність

антивірусних програм, мережевих екранів і т. д. При цьому програма оцінює кількість файлів на комп'ютері, які могли б бути «заражені» за час її перебування і роботи до моменту виявлення і видалення. Всі дані надходять на сервер, де встановлений наступний компонент розробленого комплексу програм.

Програма-сервер встановлюється на один із комп'ютерів мережі, що є первинним джерелом розповсюдження «вірусів» і має велику кількість загальних ресурсів. У даному продукті реалізований алгоритм збору інформації про стан сусідніх комп'ютерів у мережі. Програма фіксує час «зараження», рівень безпеки комп'ютера і можливість подальшого поширення «вірусу» з даного ПК.

В експерименті використовувалися методи поширення «вірусів» за допомогою копіювання в загальні папки або автозапуску з зовнішнього носія. Теоретично враховувалася можливість зараження потенційно небезпечних файлів за допомогою підрахунку їх кількості на комп'ютері: виконуваних (*.exe, *.com) і файлів, які підтримують запуск інших програм і скриптів (*.doc, *.docx, *.xls, *.xlsm, *.htm, *.bat, *.cmd). Також оцінювалася можливість доступу до розділів автозавантаження в системному реєстрі і спеціалізованих каталогів користувача.

В ході вимірювання продуктивності ЛОМ і ІС підприємств під час КНІ були отримані такі результати.

Інтенсивність вхідного потоку запитів λ розглядалася в межах від 10-8000 запитів/с. Час очікування обслуговування $\tau_{оч} = 0{,}001$ с. При інтенсивності вхідного потоку $\lambda \approx 1500$ запитів/с обслуговування запитів практично припиняється.

В ході тестів було виконано дослідження додаткових неоднорідних потоків, що формуються зараженим АРМ у складі ІС на інші АРМ, які в поточний момент часу знаходяться на зв'язку з сервером або вихідним терміналом. На рис. 6.12 - 6.17 показані результати досліджень.

На рис. 6.12 - 6.13 показані зміни якісних характеристики ІС досліджених підприємств, отримані за допомогою відповідних модулів програми «Аналізатор загроз». Зокрема, при використовуванні додаткових неоднорідних потоків (НП), як видно з отриманих графіків істотно змінилися наступні показниками ІС та АСК: загальне число зв'язків; середній час обслуговування; надійність обслуговування заявки; достовірність передачі; можливість доступу.

Аналіз отриманих результатів експериментальних досліджень дозволяє судити про закономірності можливого збільшення переданого по ЛОМ середнього та сумарного потоку заявок за рахунок додавання неоднорідних складових до потоку опрацьованих ІС запитів.

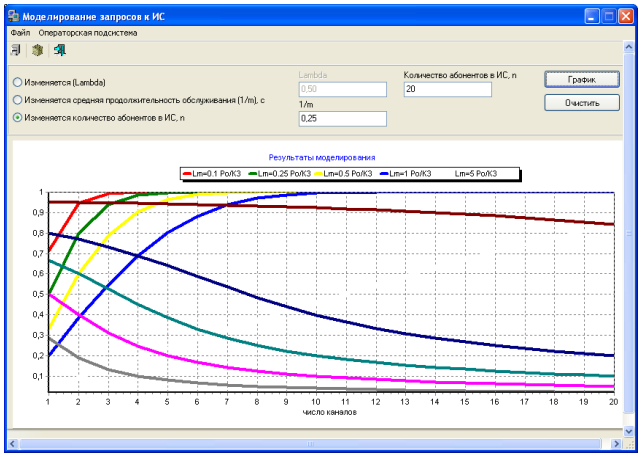


Рис. 6.12. Зміна часу обслуговування заявки в ІС підприємства при КНІ із неоднорідними потоками запитів

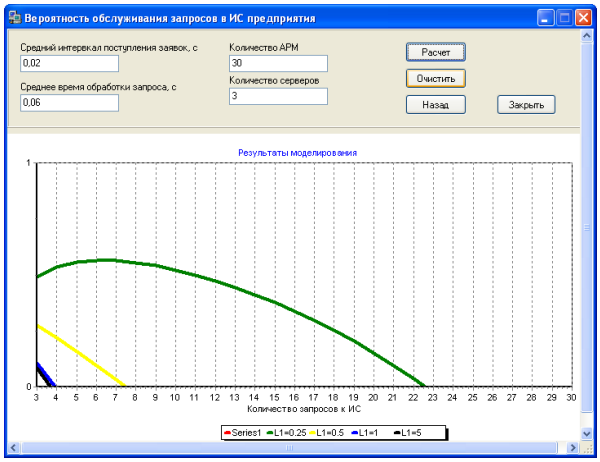


Рис. 6.13. Зміна продуктивності ІС

На рис. 6.18 представлені графіки залежності кількості «заражених» комп'ютерів від часу для розрахованих середніх оцінок безпеки комп'ютерної мережі.

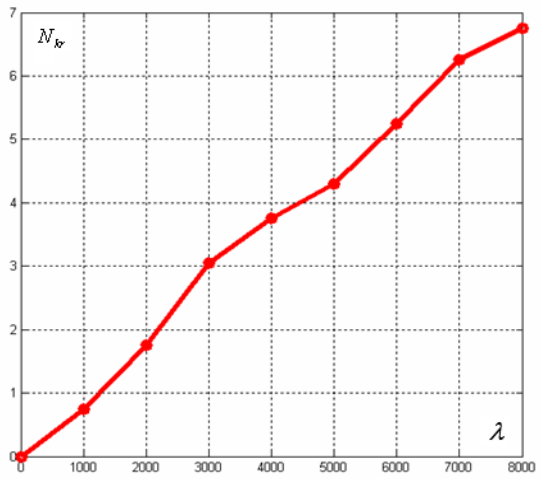


Рис. 6.14. Залежність середнього числа запитів, що очікують обслуговування, від $N_{kr}(\lambda)$

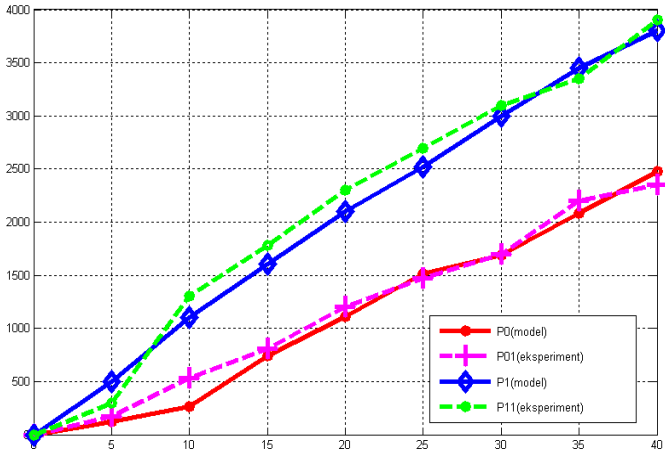


Рис. 6.15. Розподіл сумарного пуассонівського потоку запитів при розпізнаванні загрози КНІ

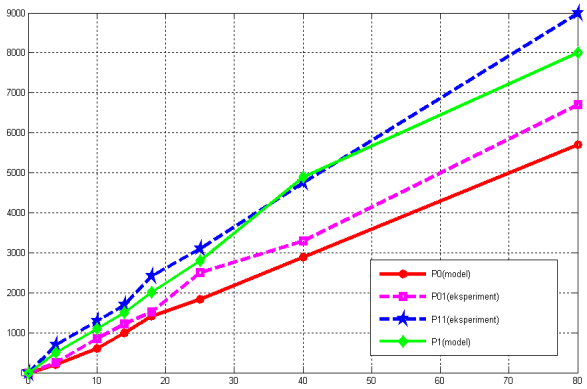


Рис. 6.16. Розподіл сумарного потоку запитів при створенні НП та розпізнаванні загрози КНІ

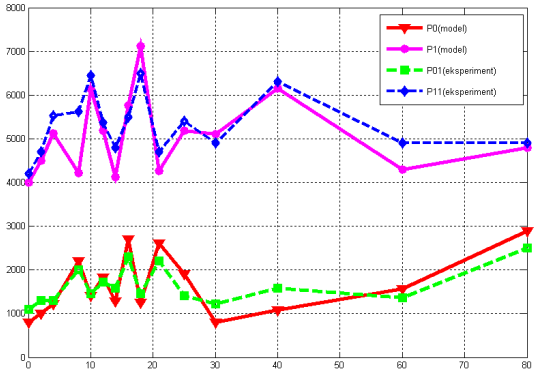


Рис. 6.17. Розподіл середнього потоку запитів при створенні НП у мережі та розпізнаванні загрози КНІ

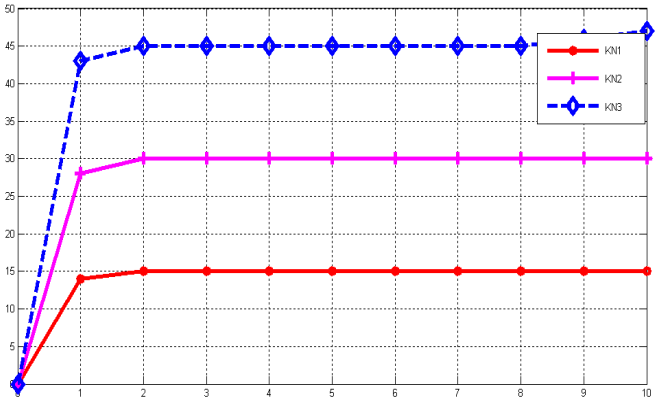


Рис. 6.18. Залежність кількості «заражених» комп'ютерів від часу для кількох середніх оцінок безпеки (середня оцінка не змінюється з перебігом часу)

В результаті проведення тестів на проникнення у ІС ТОВ «Лугавтотранс», ОЦ ДП Придніпровська залізниця, було встановлено, що:

1) при використанні експериментатором тактики присвоювання малоінтенсивному потоку даних високого пріоритету, можна збільшити в 4–6 разів середню кількість заявок у ІС за короткий проміжок часу, що 25–30 % зменшує продуктивність ІС;

2) для успішного КНІ у ІС не обов'язково створювати велику кількість запитів до сервера підприємства, а досить експлуатувати уразливості, пов'язані зі створенням малоінтенсивних пріоритетних потоків даних, зокрема, змінюючи швидкість пакета;

3) при створенні пріоритетних потоків запитів з високою інтенсивністю, час опрацювання даних у ІС підприємства збільшувався у 1,5–3 рази.

Таким чином, тестування продуктивності ІС розглянутого підприємства дозволило визначити поведінку всієї інфраструктури у звичайних умовах і при різних варіантах атак (DoS/DDoS, Probe, R2L, U2R, зараження АРМ вірусом та ін.).

Пошук уразливостей у програмному забезпеченні мережевих служб підприємства не дав позитивних результатів.

У ході тесту були зроблені контрольні запуски експлойтів на служби SMTP і WWW для перевірки можливого блокування IP-адреси, з якої проводився запуск, засобами міжмережевого екрану. Підсистема захисту не відреагувала на активні спроби атаки, адреса «порушника» заблокованою не була.

Одним із головних сервісів, що надаються ІС компанії користувачам глобальної мережі Інтернет, є попереднє оформлення замовлення на перевезення вантажу. Також в ІС розглянутого підприємства передбачена можливість автоматичного розміщення на серверах підприємств динамічних WEB-сторінок, що містять періодично обновлювану інформацію комерційного характеру (тарифи на перевезення, відомості для попутного перевезення

вантажів та ін.) В ході тестів на проникнення досліджувалася можливість використання SQL – ін'єкцій для зміни такої комерційної інформації [4-15].

Отже, результати проведеного тесту на проникнення дозволили зробити такі висновки:

1) зовнішній периметр ІС підприємств захищений достатньо надійно. Регулярно виконується установка оновлень програмного забезпечення мережевих служб, конфігурація служб відповідає вимогам ІБ. Тим не менше, мережеві служби надають потенційному порушнику достовірну службову інформацію, що може бути використана при організації атак на зовнішній периметр;

2) система виявлення КНІ установлена в конфігурації за замовчуванням, її налаштування неефективне і не забезпечує адекватний рівень реакції на явно виражену мережеву активність порушника;

3) загальна архітектура ІС підприємства, конкретні технічні рішення щодо забезпечення інформаційної безпеки і недостатня кваліфікація користувачів ІС не забезпечують необхідний рівень захисту; рівень захищеності серверів та АРМ ІС – низький;

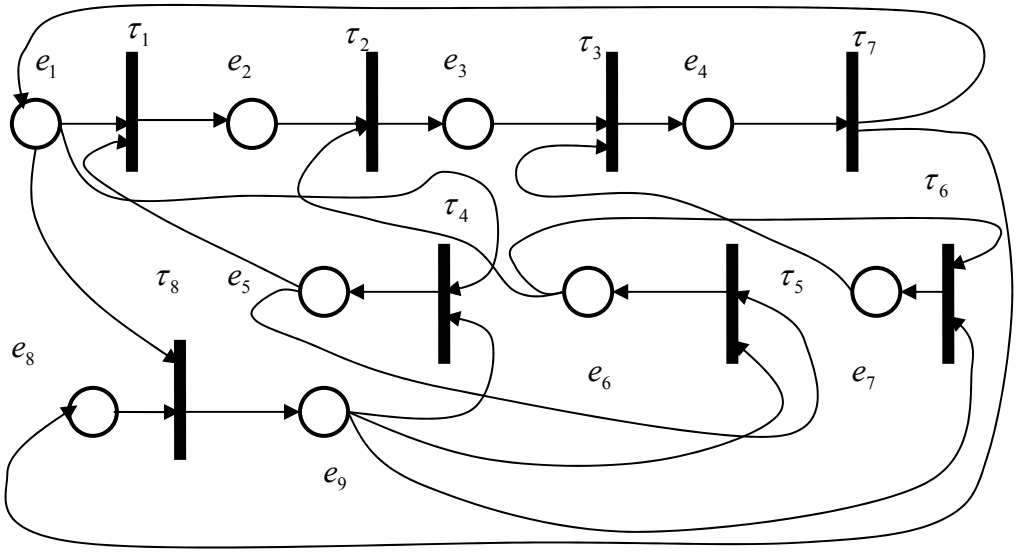
4) для роботи з сервером СУБД варто використовувати нестандартні порти;

5) при експлуатації модулів ІС, що мають вихід в Інтернет, необхідно заборонити спільне використання по ЛОМ тих каталогів, які містять файли баз даних підприємства.

Для посилення ступеня захисту інформації, керівникам підприємства запропоновано використовувати безперервне коригування профілів активних користувачів, зокрема так званий ітераційний алгоритм (ІА). Значення ітераційного алгоритму полягає в неявному зворотному зв'язку сервера з користувачем, що реалізується через облік статистики запитів. Отримана оцінка поточного профілю користувача використовується для ранжирування користувачів на групи за ступенем небезпеки для ресурсів ІС: а) користувач; б) потенційно небезпечний користувач; в) небезпечний користувач; г) пору-шник.

Для синтезу процедури автоматичної класифікації застосований апарат дискретних процедур розпізнавання загроз та пошуку уразливостей, детально описаний в роботах [4-15].

На рис. 6.19 представлена схема, що відображає логічну структуру моделі системи керування правами доступу, побудованої із використанням мереж Петрі. Позиції відображають функціональні стани системи управління доступом.



Умовні позначення: e_1 - активний стан користувача; e_2 - користувач допущений до роботи з документами ІС (договори, контракти та ін.); e_3 - користувач допущений до роботи з функціональними компонентами ІС ТП; e_4 - користувач допущений до файлів інформаційного ресурсу; e_5 - перевірка прав користувача щодо роботи з документами; e_6 - перевірка прав доступу користувача до функціональних компонентів ІС ТП; e_7 - перевірка прав користувача щодо файлів; e_8 - відновлення початкового стану системи управління доступом; e_9 - обмеження активності користувача в часі, що пов'язано з розробкою або коригуванням інформаційного ресурсу.

Рис. 6.19. Схема моделі адаптивного управління правами доступу в ІС

Розроблена імітаційна модель адаптивного рольового управління правами доступу до інформаційних ресурсів ТОВ «Лугавтотранс», ОЦ ДП Придніпровська залізниця, дала змогу відобразити логіко-ймовірнісні зв'язки та інформаційні процеси, що протікають в ІКС. Запропоновані моделі в сукупності дозволяють виявляти потенційні уразливості в ході реалізації

окремих етапів тесту на проникнення і здійснювати настроювання механізмів захисту ІС.

В таблиці 6.4 наведені результати експериментальних досліджень з інтелектуального розпізнавання загроз ІС та АСК підприємств ТОВ «Лугавтотранс», ОЦ ДП Придніпровська залізниця (ПЗ), а також порівняння із результатами імітаційного моделювання, розглянутого у попередньому розділі роботи.

Таблиця 6.4

Результати експериментальних досліджень з інтелектуального розпізнавання загроз ІС та АСК підприємств у порівнянні імітаційними моделями

№	Параметр	ТОВ «Луг-автотранс»	ОЦ ДП «ПЗ»	Імітаційна модель	Відхилення, %
1	Виявлено атак за типами, %: DoS/DDoS Probe R2L U2R	 92-98 88-95 49-50 67-70	 91-95 90-94 42-45 65-69	 85-96 - - -	7-9
2	Ймовірність досягнення зловмисником мети за $T_{зад}=60$ хвилин: DoS/DDoS Probe R2L U2R	 0,10–0,11 0,08–0,11 0,06–0,07 0,04–0,06	 0,06–0,092 0,05–0,067 0,02–0,04 0,02–0,05	 0,08–0,1	0,05–0,1
3	Час отриманням доступу незареєстрованого користувача до комп'ютера з боку віддаленої машини (атака R2L), год	4–5	7–9	-	-
4	Час отриманням зареєстрованим користувачем права адміністратора (атака U2R), год	1–1,5	1–2	-	-
5	Час потрібний на зміну інтенсивності запитів та їхнього пріоритету для ІС або АСК, год	0,5–1	0,5–1	0,9–1,2	5–10
6	Сумарна кількість запитів при створенні неоднорідних потоків у системі, запитів/с	4000–10000	6000–15000	4000–12000	10-20
7	Середня кількість запитів при створенні неоднорідних потоків у системі, запитів/с	6000–8000	8000–10000	4000–9000	10–20

До проведення тестування на проникнення не варто підходити як до завдання, що виконується один раз. В умовах функціонування сучасних підприємств ІТ-інфраструктура може дуже швидко розвиватися і відповідно змінюватися. Змінюються властивості мережі, склад і якість прикладних сервісів, число користувачів, властивості СЗІБ. Тому оцінка реакції системи на підвищення навантаження повинна бути регулярною і виконуватися, наприклад, при проведенні періодичного аудиту ІБ.

6.3. Висновки до розділу 6

Результатом проведених у даному розділі навчального посібника стали такі висновки.

1. Проведено тести на проникнення (за погодженням з керівництвом) в ІС та АСК підприємств ТОВ «Лугавтотранс» та ОЦ ДП Придніпровська залізниця.

2. Виконано тестування продуктивності ЛОМ і ІС підприємств, а також досліджено поведінку всієї інфраструктури в звичайних умовах і при різних варіантах КНІ.

3. Експериментально підтверджена ефективність запропонованого методу та моделі інтелектуального розпізнавання КНІ у ІС при використанні потоків заявок із різною інтенсивністю надходження, у разі експлуатації уразливостей, пов'язаних зі створенням малоінтенсивних пріоритетних потоків.

4. Виконано тестування розробленої експертної системи з інтелектуального розпізнавання загроз ІБ підприємств, а також досліджено поведінку всієї інфраструктури в звичайних умовах і при різних варіантах КНІ (комп’ютерні атаки, зараження АРМ вірусом тощо). Експериментально підтверджена можливість ефективного розпізнавання загроз ІБ (з точністю до 85-98 %) при реалізації складних КНІ у модулі e-business ІС та АСК при використанні потоків заявок із різною інтенсивністю надходження, у разі

експлуатації уразливостей, пов'язаних зі створенням малоінтенсивних пріоритетних потоків.

5. Виявлено, що серед загроз ІБ підприємств на яких проводився тест на проникнення, найбільш суттєвими є: використовування застарілого ПЗ; адміністративні та технологічні труднощі для оновлення ПЗ та СУБД; доступ сторонніх компаній до технологічних мереж. В системах SCADA, загрозами ІБ є поширені уразливості ПЗ до атак DoS/DDoS та SQL Injection.

6. Розроблено модель адаптивного рольового управління правами доступу до інформаційних ресурсів підприємств, що дозволяє виявляти потенційні уразливості в ході реалізації окремих етапів тесту на проникнення і здійснювати настроювання механізмів захисту ІКС.

7. Протестована розроблена експертна система «Аналізатор загроз», яка дозволяє вирішувати такі завдання: розпізнавання загроз ІБ, збір інформації про стан комп'ютерів у мережі підприємства; сканування запущених програм і процесів на АРМ підприємства; сканування доступних портів; визначення рівнів безпеки комп'ютера (АРМ) і можливість подальшого поширення «вірусу» з даного АРМ; фіксації часу «зараження» комп'ютерів у мережі; оцінка поточних ризиків НСД до ІС підприємства; моделювання процесу ухвалення рішення по вибору оптимальної стратегії управління інвестиційним проектуванням СЗІ для господарюючого суб'єкта.

ВИСНОВКИ

У навчальному посібнику розв’язано важливу науково-прикладну проблему – створення теоретичних засад, методів, моделей та програмних продуктів для забезпечення інформаційної безпеки за умови збільшення кількості дестабілізуючих впливів на схоронність і цілісність інформації.

У навчальному посібнику:

1. Розглянуті питання впровадження сучасних інформаційно-комунікаційних систем і технологій. З’ясовано, що складність застосування до систем розпізнавання загроз формалізованого апарату аналізу й синтезу СЗІ ІКС полягає в тому, що конкретний інформаційний комплекс і його підсистема ІБ складаються з різнорідних елементів, які описуються із використанням різних математичних моделей. Показано, що застосування елементів адаптивного захисту інформації може бути засноване на використанні новітніх методів і моделей інтелектуального розпізнавання загроз ІКС.

2. Викладено метод інтелектуального розпізнавання загроз на основі дискретних процедур із використанням апарату логічних функцій та нечітких множин, що дозволяє підвищити ефективність розпізнавання загроз ІКС залежно від класу до 85–98%, створювати ефективні аналітичні, схемотехнічні та програмні рішення СЗІ ІКС.

3. Викладено метод складання вирішального правила для дискретних процедур розпізнавання загроз ІБ, який дозволяє виконувати інтелектуальне розпізнавання загрози з мінімальною кількістю помилок. Показано, що побудова множини елементарних класифікаторів для розглянутих класів загроз зводиться до знаходження припустимих і максимальних кон’юнкцій для характеристичної функції класу.

4. Установлено, що в моделі «голосування» за представницькими наборами при вирішенні завдань інтелектуального розпізнавання загроз ІКС досить обмежитися побудовою представницьких наборів довжини 3. З’ясовано, що ДПРЗ ІБ чутливі до наявності «шуму» в тренувальному наборі. Для

подолання цих недоліків запропоновано використовувати математичний апарат нечітких множин.

Визначено, що в завданні інтелектуального розпізнавання комп'ютерних нападів на інформацію КНІ у ІКС частина значень ознак мають вагу, близьку до нуля, але при цьому багато і таких значень, які мають більшу вагу, тобто є дуже типовими для одного із класів загроз для інформаційної безпеки.

5. Викладено математичну модель функціонування СЗІ ІКС при неоднорідних потоках вимог і мережних класах загроз. Встановлено, що марковські моделі процесів широко використовуються при аналізі й синтезі СЗІ ІКС, причому властивість марковості є певним обмеженням на використовувані реальні сигнали, але цілком достатнім для розробки змістовних методів аналізу й синтезу комплексів СЗІ.

Визначено, що математичні моделі із використанням апарату ланцюгів Маркова є ефективним інструментом для кількісної оцінки можливості реалізації КНІ у ІКС.

6. Викладено моделі програмних атак на ІКС, які, на відміну від існуючих, враховують неоднорідні вхідні потоки запитів та апріорі виділяють найбільш інтенсивні вхідні потоки і враховують наявність черг за потоками, що дозволяє проводити їх дослідження, здійснювати вибір способів протидії та нейтралізацію наслідків від їх впливу, аналізувати більш складні і раніше невідомі види програмних атак.

Удосконалено моделі інтелектуального розпізнавання загроз ІКС з урахуванням можливостей зміни нападаючими інтенсивності неоднорідних потоків запитів. Досліджено динаміку станів СЗІ ІКС у разі кількох неоднорідних потоків вимог.

Викладено послідовність етапів моделювання роботи комплексів СЗІ ІКС за умови впливу неоднорідних потоків даних.

7. Визначено, що використання імітаційного моделювання, яке об'єднує між собою різнорідні математичні моделі елементів, що входять до складу ІКС,

є одним із методів, які дозволяють оцінити ефективність СЗІ та її реакцію на спроби НСД (збурення) за рядом показників.

Встановлено, що за допомогою імітаційного моделювання в середовищі MATLAB та Simulink при створенні ЗЗІ ІКС можуть вирішуватися завдання з визначення шляхів удосконалення системи захисту інформації на підставі аналізу різних варіантів технічної, технологічної, а також організаційної перебудови та дослідження наслідків прийнятих рішень.

Проаналізовано можливість написання і підключення власних модулів для реалізації математичних моделей, що описують ймовірнісні стани системи та її ентропію. Показана можливість зміни параметрів імітаційного моделювання під час проведення експериментів в MATLAB та Simulink. Встановлено, що програмна реалізація математичних моделей ЗЗІ для ІКС дозволяє отримати наочні уявлення процесів у ДЄІІ та ІКС. Визначено важливі характеристики СМО (час простою лінії, середній час передачі даних по лінії від кожної станції і т. п.). Реалізовано в MATLAB 7/2009 та Simulink імітаційну модель інтелектуального розпізнавання загроз (ДПРЗ ІБ) із використанням апарату логічних функцій і нечітких множин ознак НСД у ІКС. Розроблені імітаційні моделі КНІ дозволяють на 25–30% зменшити час налагодження проектів СЗІ ІКС та АСК.

8. Викладено нові підходи до вирішення завдання оптимізації складу обраної структури ЗЗІ ІКС за різними критеріями. Запропоновано використовувати показник поточних ризиків несанкціонованого доступу в систему для оцінки в реальному масштабі часу загроз ІБ, зокрема, з урахуванням поточних значень неоднорідних потоків даних. Проаналізовано (із використанням ігрових моделей) ймовірності вибору зловмисником тієї чи іншої загрози (класу загрози) для подальшої реалізації. Отримано залежності для моделювання станів ІКС та АСК при реалізації розглянутих класів загроз із використанням напівмарковського процесу.

Формалізовано завдання оптимізації складу комплексів ЗЗІ ІКС за такими критеріями: мінімум ймовірності досягнення порушником усіх цілей; мінімум

середнього рівня втрат ІКС та АСК від реалізації порушником усіх цілей; максимум ймовірності успішної протидії ЗЗІ реалізації всіх цілей порушником; мінімум значення інтегрального показника «вартість-ризик».

9. Запропоновано вирішувати завдання визначення оптимального складу комплексу ЗЗІ ІКС за критерієм максимуму ймовірності успішної протидії СЗІ реалізації всіх цілей порушником. Запропоновано математичну модель оптимізації структурно-технологічного резерву критично важливого програмного забезпечення ІКС. Розглянуто особливості застосування методів дискретної оптимізації для вирішення завдань забезпечення захисту критичної інформації ІКС. Запропоновано алгоритм оцінки меж рішенням двоїстої задачі з визначенням порядку розгалуження для вирішення завдань забезпечення захисту інформації в ІКС. Аналіз результатів показав, що застосування способу попереднього визначення порядку розгалуження змінних разом із симплекс-методом оцінки меж рішення дозволяє скоротити час вирішення завдань в 5–20 разів.

10. Встановлено, що використання розробленого комплексу моделей, методів і алгоритмів оптимізації інформаційно-обчислювального процесу і забезпечення схоронності й захищеності інформації дає змогу підвищити обґрунтованість прийнятих рішень на етапах проектування ІКС експлуатації та реконструкції ІКС з розподіленим опрацюванням. Зокрема, обсяг інформації, яка циркулює в ІКС, за рахунок раціонального розподілу ПМ та ІМ зменшено на 17–20%, при цьому більш ніж на 35% зросла ймовірність вирішення всіх завдань в системі за мінімального обсягу відновного резерву кожного ПМ та ІМ, одночасному збільшенні стійкості інформаційно-обчислювального процесу.

Ймовірність вирішення всіх завдань у системі з урахуванням відновного резервування при обсязі резерву в одну копію зросла на 28–30% при збільшенні ймовірності вирішення кожного із завдань не нижче ніж на 5–7%.

11. Розроблено експертну систему «Аналізатор загроз», яка дозволяє вирішувати такі завдання: розпізнавання загроз ІБ, збір інформації про стан комп'ютерів у мережі підприємств; сканування запущених програм і процесів

на АРМ та доступних портів ІС; визначення рівнів безпеки АРМ і можливість подальшого поширення «вірусу» з АРМ; оцінювання поточних ризиків НСД до ІС підприємств; моделювання процесу ухвалення рішення щодо вибору оптимальної стратегії управління інвестиційним проектуванням СЗІ ІКС.

Представлені у навчальному посібнику результати досліджень дозволяють забезпечувати захист ІКС з урахуванням впливу засобів і методів захисту інформації на функціональні характеристики ІКС.

Результати даних досліджень можуть бути використані в науково-дослідних організаціях фахівцями в галузі створення та організації функціонування ІКС, АСК, розробки ефективних алгоритмів у сфері захисту інформації, а також студентами та аспірантами вищих навчальних закладів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Kirilchuk I., Barkov A. Methodological aspects of the development of automated information and analytical waste management systems //International Multidisciplinary Scientific GeoConference: SGEM. – 2016. – Т. 2. – С. 577-582.

2. Lakhno V., Petrov A.S.The information protection in automated system on transport: monograf AGH University of Science and Technology, Krakow, Poland. – Kra-kow, Lugansk: Knowledge press, 2012. – 169 p.

3. Комплексні системи захисту інформації : навчальний посібник / [Яремчук Ю. Є., Павловський П. В., Катаєв В. С., Сінюгін В. В.].Вінниця : ВНТУ, 2018. - 118 с.

4. Проєктування комплексних систем захисту інформації. Підручник / В. О. Хорошко, І. М. Павлов, Ю. Я. Бобало, В. Б. Дудикевич, І. Р. Опірський, Л. Т. Пархуць. Львів : Видавництво Львівської політехніки, 2020. 320 с

5. Технічні канали витоку інформації. Порядок створення комплексів технічного захисту інформації. Навчальний посібник / Іванченко С.О., Гавриленко О.В., Липський О.А., Шевцов А.С. – К.: ІСЗЗІ НТУУ «КПІ», 2015. – 104 с.

6. Методи та засоби інженерно-технічного захисту інформації навч. посіб. / В.В. Богданов, О.В.Волков, О.В.Жук, В.В.Мартинюк – К.: ВІТІ НТУУ «КПІ», 2013.

7. Petrov O., Borowik B., Karpinskyy M, Korchenko O., Lakhno V. Immune and defensive corporate systems with intellectual identification of threats. Pszczyna : Śląska Oficyna Drukarska, 2016, P. 222.

8. Методи та засоби захисту інформації [Навчальний посібник] / В.А. Лахно, Є.В. Васіліу, В.М. Гладких, В.М. Домрачев, Н.М. Сивкова. – К. : ЦП «Компринт» О.В., 2020. – 444 с.

9. Лахно В.А., Блозва А.І., Касаткін Д.Ю. навчальний посібник «Робототехнічні комп'ютерні системи» / В.А.Лахно, А.І.Блозва, Д.Ю.Касаткін // НУБіП України, - Київ, Видавничий центр Компринт 2021, 169 с.

10. Сагун А.В., Лахно В.А., Бобков В.Б., Касаткін Д.Ю., Хайдуров В.В. навчальний посібник «Спеціалізовані комп'ютери» / А.В.Сагун, В.А.Лахно, В.Б.Бобков, Д.Ю.Касаткін, В.В.Хайдуров // НУБіП України, - Київ, Видавничий центр Компринт 2021, 190 с.

11. Lakhno, V., Satzhanov, B., Tabylov, A., Chubaievsyi, V., Kaminskyi, S. Organizational and Economic Provision of Corporate Information Effective Protection (2023) CEUR Workshop Proceedings, 3421, pp. 138-147.

12. Lakhno, V., Mazaraki, A., Kasatkin, D., Kryvoruchko, O., Khorolska, K., Chubaievskyi, V. Models and Algorithms for Optimization of the Backup Equipment for the Intelligent Automated Control System Smart City (2023) Lecture Notes in Networks and Systems, 383, pp. 749-762.

13. Lakhno, V., Akhmetov, B., Smirnov, O., Chubaievskyi, V., Khorolska, K., Bebeshko, B. Selection of a Rational Composition of İnformation Protection Means Using a Genetic Algorithm (2023) Lecture Notes on Data Engineering and Communications Technologies, 131, pp. 21-34.

14. Lakhno, V., Kasatkin, D., Desiatko, A., Chubaievskyi, V., Tsuitsuira, S., Tsuitsuira, M. Indicators Systematization of Unauthorized Access to Corporate Information

15. (2023) Lecture Notes on Data Engineering and Communications Technologies, 131, pp. 569-580.

Додаток А

Приклад лістингу m-файлу для функції hblk, яка описує комутатор мережі ІКС

```
function hblk = firsrccommutator(hTar, name,RCF,qparam, dspblklibname,render)
error(nargchk(5,6,nargin,'struct'));
RCF = str2num(RCF);
L = RCF(1);
M = RCF(2);
for i = 0:L-1
   phasesel(i+1) = mod(M*i,L);
end
if nargin<6
   render=true;
end
if ~render
   return
end
sys = hTar.system;
idx = findstr(sys, '/');
set_param(0,'CurrentSystem',sys(1:idx(end)-1));
hTarFirsrcCommutator = dspfwiztargets.dgtarget;
hTarFirsrcCommutator.destination = 'current';
idx = findstr(sys,'/');
if length(idx) == 1
   blockpath = hTar.blockname;
else
   blockpath = sys(idx(end)+1:end);
end

p = positions;
hTarFirsrcCommutator.blockname = [blockpath '/' name];
pos = createmodel(hTarFirsrcCommutator);
hsubsys = add_block('built-in/subsystem',hTarFirsrcCommutator.system,
'Tag','FilterWizardSampleRateConverterCommutator');
set_param(hsubsys,'Position',pos);
subsys = hTarFirsrcCommutator.system;
L_str = num2str(L);
blkname = 'phaseselector';
hblk = hTarFirsrcCommutator.repeatingsequencestair(blkname);
set_param(hblk,'Position',[185 35 215 65]);
set_param(hblk,'OutValues',strcat('[',num2str(phasesel),']'));
blkname = 'multiswitch';
hblk = hTarFirsrcCommutator.multiportswitch(blkname,L_str,'on');
set_param(hblk,'Position',[280 30 370 L*min(100,32000/L)+100+30]);
add_line(subsys,'phaseselector/1','multiswitch/1','autorouting','on');
for m = 1:L
   inblkname = ['input',num2str(m)];
   hblk = hTarFirsrcCommutator.inport(inblkname,qparam);
   set_param(hblk,'Position',inportpos(m,p,L));
```

```
   add_line(subsys,[inblkname '/1'],['multiswitch/',num2str(m+1)],'autorouting','on');
end
blkname = 'output';
hblk = outport(hTarFirsrcCommutator,blkname);
set_param(hblk,'Position',[435 50*L+35 465 50*L+65]);
add_line(subsys,'multiswitch/1','output/1','autorouting','on');
function p = positions
p.input = [-15 -8 15 8];
function pos = inportpos(stage,p,N)
pos = min(32000/N,100)*[0 stage 0 stage]+[50 50 50 50]+p.input;
hFig = ancestor(hcbo, 'figure');
if isappdata(hFig, 'DataCursorManager')
   hDCM = getappdata(hFig, 'DataCursorManager');
else     hDCM = graphics.datacursormanager(hFig);
   setappdata(hFig, 'DataCursorManager', hDCM);
end
% This is a workaround to resolve the datacursor bug in pole/zero plots.
set(hDCM, 'EnableZStacking', 0);
hB = hgbehaviorfactory('datacursor');
if nargin < 3,
   hB.UpdateFcn = @stringFcn;
else     hB.UpdateFcn = dataMarkerFcn;
end
hgaddbehavior(hcbo,hB);
h = hDCM.createDatatip(hcbo);
h.UIContextMenu = uicontextmenu('Parent',ancestor(hcbo,'hg.figure'));
datacursormenus(hDCM,'alignment','fontsize','movable','interpolation', 'export','delete','deleteall');
function dataTip = stringFcn(hLine, eventData)
hAx  = ancestor(hLine, 'axes');
hxlbl = get(hAx,'Xlabel'); xlbl = get(hxlbl,'String');
hylbl = get(hAx,'Ylabel'); ylbl = get(hylbl,'String');
xlbl = localTrimBrackets(xlbl);
ylbl = localTrimBrackets(ylbl);
dataTip = sprintf('%s: %.7g\n%s: %.7g', xlbl, eventData.Position(1), ...
   ylbl, eventData.Position(2));
function output = localTrimBrackets(input)
idx = findstr(input, '(');
if ~isempty(idx)
   output = strtrim(input(1:idx-1));
else
   output = strtrim(input);
end
```

Додаток Б

Приклад лістингу m-файлів для функцій commwlan80211a_settings і commwlan80211a_udg, що описують сегмент ЛОМ ІКС

```
%-------------------------------------------------------------------------
function commwlan80211a_settings
persistent postloadFlag;
if isempty(postloadFlag)
   postloadFlag = true;
else
   if postloadFlag
      postloadFlag = false;
      return
   end
end
settingsBlock = [bdroot '/Model Parameters'];
[OFDMSymPerFrame, ...
 OFDMTrainPerFrame, ...
 vtbd, ...
 thres, ...
 hyst] ...
 = getSettings(settingsBlock, ...
  'OFDMSymPerFrame', ...
  'OFDMTrainPerFrame', ...
  'vtbd', ...
  'SNR0_dB', ...
  'hyst');
if mod(OFDMSymPerFrame, 2)~=0
   error('commblks:commwlan80211a_settings:InvalidSymbolsPerFrame','OFDM symbols
per frame must be even.');
end

p.NSD = 48;  % number of data symbols in OFDM symbol
p.NST = 52;  % number of data symbols and pilots in OFDM symbol
% Note: NSD and NST are same as symbols in 802.11a standard.
p.NFFT = 64; % number of points on FFT
p.NcyclicPrefix = 16;
p.NFFT2 = p.NFFT + p.NcyclicPrefix;
p.TXFFTShiftIndices = [p.NST/2+1:p.NFFT 1:p.NST/2];
p.TXCyclicPrefixIndices = [p.NFFT-[p.NcyclicPrefix-1:-1:0] 1:p.NFFT];
p.RXCyclicPrefixIndices = [p.NcyclicPrefix+1:p.NFFT2];
p.RXSelectFFTIndices = [p.NFFT-[p.NST/2-1:-1:0] 1:p.NST/2+1];
p.OFDMSymPerFrame = OFDMSymPerFrame;
p.OFDMTrainPerFrame = OFDMTrainPerFrame;
p.OFDMTotSymPerFrame = OFDMSymPerFrame + OFDMTrainPerFrame;
p.numTxSymbols = p.NSD * OFDMSymPerFrame;
p.numTrainingSymbols = p.NSD * OFDMTrainPerFrame;
p.long_training_seq = ...
   [1 1 -1 -1 1 1 -1 1 -1 1 1 1 1 1 1 -1 -1 1 1 -1 1 -1 1 1 1 1 1 0 ...
    1 -1 -1 1 1 -1 1 -1 1 -1 -1 -1 -1 -1 1 1 -1 -1 1 -1 1 -1 1 1 1 1 1].';
p.numModulators = 8;
```

```
        p.txBitsPerSymbol = [1 1 2 2 4 4 6 6];
        p.txBitsPerBlock = p.numTxSymbols * p.txBitsPerSymbol;
        p.modOrder = 2.^p.txBitsPerSymbol;
        p.codeRate = [1/2 3/4 1/2 3/4 1/2 3/4 2/3 3/4];
        p.bitsPerBlock = p.txBitsPerBlock .* p.codeRate;
        p.bitsPerSymbol = p.txBitsPerSymbol .* p.codeRate;
        p.maxBitsPerBlock = max(p.bitsPerBlock);
        p.nSource = min( gcd( min(p.bitsPerBlock), p.bitsPerBlock ) );
        p.nS = p.bitsPerBlock/p.nSource;
        p.OFDMSymbolPeriod = 4e-6;
        p.OFDMPilotPeriod  =
p.OFDMSymbolPeriod*p.OFDMTotSymPerFrame./p.OFDMSymPerFrame;
        p.blockPeriod      = p.OFDMTotSymPerFrame.*p.OFDMSymbolPeriod;
        p.bitPeriod        = p.blockPeriod./ p.bitsPerBlock;
        p.minBitPeriod     = min(p.bitPeriod);
        p.chanSamplePeriod = p.blockPeriod/(p.OFDMTotSymPerFrame * p.NFFT2);
        vtbd_set = vtbd(ones(1, p.numModulators));
        p.vtbd_set = vtbd_set;
        p.link_delay = vtbd_set;
        p.thres = thres;
        p.hyst = hyst;
        assignin('base', 'params', p);

        %-----------------------------------------------------------------------
        function varargout = getSettings(settingsBlock, varargin)
        h = get_param(settingsBlock, 'handle');
        for n = 1:length(varargin)
           varargout{n} = evalin('base', get(h, varargin{n}));
        end

        function [sys, x0, str, ts] = commwlan80211a_udg(t, x, u, varargin)

        switch varargin{1}
           case 3
              sys = [];  % mdlOutput - unused
           case 2
              sys = mdlUpdate(t, x, u, varargin{:});
           case 0
              [sys, x0, str, ts] = mdlInitializeSizes;
           case 9
              sys=mdlTerminate;
           otherwise
              feval(varargin{:});
        end


        % ---------------------------------------------------------
        function [sys, x0, str, ts] = mdlInitializeSizes
        blk = get_param(gcb, 'parent');
        sizes = simsizes;
        sizes.NumInputs      = -1;
        sizes.NumOutputs     =  0;
        sizes.NumSampleTimes =  1;
```

```
sys = simsizes(sizes);
x0  = [];          % states
str = '';          % state ordering strings
ts  = [-1 1];      % inherited sample time, fixed in minor steps
% initialize block data
bd = get_param(blk, 'userdata');
bd.firstcall = true;
bd.figTag = blk;
bd.graphicsName = get_param(blk, 'graphicsName');
bd.plotFcnHandle = str2func(bd.graphicsName);
bd.cellArrayMode = strcmp(get_param(blk, 'convertMode'), 'cell array');
set_param(blk, 'userdata', bd);
% ------------------------------------------------------------
function sys = mdlUpdate(t, x, u, flag, params)
% Faster implementation of: blk=gcb;
cs   = get_param(0, 'CurrentSystem');
cb   = get_param(cs, 'CurrentBlock');
sfcn = [cs '/' cb];
blk  = get_param(sfcn, 'parent');
sys  = [];
bd = get_param(blk, 'userdata');
bd.fig = findobj('type', 'figure', 'tag', bd.figTag);
if isempty(bd.fig)
   % figure is not open
     bd.firstcall = true;
 else
   % figure is open
   if bd.firstcall
      % get axes handles
      [bd.axes, bd.multiple_axes] = get_axes_handles(bd.fig);
   end

   [M, N] = size(u);
   u_complex = u(:, 1:N/2) + j*u(:, N/2+1:N);

   input_names = params{1};
   other_params = params{2:length(params)};
   other_params_exist = ~isempty(other_params);
   plot_data = convert_simulink_vector(u_complex, input_names, bd.cellArrayMode);
   if other_params_exist
      feval(bd.plotFcnHandle, plot_data, bd.axes, bd.firstcall, other_params);
   else
      feval(bd.plotFcnHandle, plot_data, bd.axes, bd.firstcall)
   end

   bd.firstcall = false;

end

set_param(blk, 'userdata', bd);
% ------------------------------------------------------------
function sys = mdlTerminate
sys = [];
```

```
%--------------------------------------------------------------------------
function data = convert_simulink_vector(u, input_names, cellArrayMode);

num_inputs = length(input_names);
if num_inputs == 1
   data = u(2:end, :);
   return
end
length_u = length(u);
if cellArrayMode
   data = cell(1, num_inputs);
else
   data = cell2struct(cell(1, num_inputs), input_names, 2);
end

idx = 1;
for i = 1:num_inputs
   L = u(idx);
   v = u(idx+1 : idx+L, :);
   if cellArrayMode
      data{i} = v;
   else
      data = setfield( data, input_names{i}, v );
   end
   idx = idx + L + 1;
end
%--------------------------------------------------------------------------
function [haxes, multi] = get_axes_handles(fig);
c = get(fig, 'children');
multi = (length(c)>1);
if ~multi
   haxes = c;
else
   t = get(c, 'tag');
   h = num2cell(c);
   x = [t(:).'; h(:).'];
   haxes = struct(x{:});
end
```

Додаток В

Лістинг програми «Аналізатор загроз»

```
unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ComCtrls, CommCtrl, Winsock, ImgList, ShellAPI, XPMan,
  jpeg, ExtCtrls, FileCtrl, AdvProgressBar, inifiles, registry,  IpHlpApi, IpTypes,  IpIfConst,
  Grids, AdvObj, BaseGrid, AdvGrid, AdvPanel, AdvGlowButton,
  AdvGridWorkbook, Buttons, rtflabel, TeeProcs, TeEngine, Chart, Series;
resourcestring
RES_THREADCOUNT = 'Запущено потоків: %d';
   RES_COMPCOUNT = 'Знайдено: %d';
   RES_ERR_RANGE = 'Неприпустимий діапазон';
const
  TH32CS_SNAPPROCESS  = $00000002;
  //
  MIB_TCP_STATE_CLOSED    = 1;
  MIB_TCP_STATE_LISTEN    = 2;
  MIB_TCP_STATE_SYN_SENT  = 3;
  MIB_TCP_STATE_SYN_RCVD  = 4;
  MIB_TCP_STATE_ESTAB     = 5;
  MIB_TCP_STATE_FIN_WAIT1  = 6;
  MIB_TCP_STATE_FIN_WAIT2  = 7;
  MIB_TCP_STATE_CLOSE_WAIT = 8;
  MIB_TCP_STATE_CLOSING    = 9;
  MIB_TCP_STATE_LAST_ACK   = 10;
  MIB_TCP_STATE_TIME_WAIT  = 11;
  MIB_TCP_STATE_DELETE_TCB = 12;
  STR_START    =  ' Почати сканування';
  STR_STOP     =  ' Зупинити сканування';
  STR_STARTED  =   Йде сканування... ...';
  STR_STOPPED  =  ' Сканування завершено...';
  STR_END      =  ' Завершення потоку...';
  STR_FIELD    =  'Не обрано ...';
  WSA_TYPE = $101;
    RES_UNKNOWN = ' Невідомо';
    RES_IP      = 'IP адреса: ';
    RES_CMP     = ' Ім'я комп'ютера: ';
    RES_USR     = ' Ім'я користувача: ';
    RES_DOM     = 'Домен: ';
    RES_SER     = 'Сервер домену: ';
    RES_COM     = 'Коментар: ';
    RES_PROV    = 'Провайдер: ';
    RES_GRP     = 'Групи: ';
    RES_MAC     = 'MAC адреса: ';
    RES_SHARES  = 'Доступні ресурси: ';
    RES_TIME    = 'Затрачено часу: ';
    RES_COM_NO  = 'Відсутні';
  IPHLPAPI = 'IPHLPAPI.DLL';
  MAX_ADAPTER_ADDRESS_LENGTH = 7;
```

```
type
    //---------------------------------------------
  PTMibTCPExRow = ^TMibTCPExRow;
  TMibTCPExRow = packed record
    dwState: DWORD;
    dwLocalAddr: DWORD;
    dwLocalPort: DWORD;
    dwRemoteAddr: DWORD;
    dwRemotePort: DWORD;
    dwProcessID: DWORD;
  end;
  PTMibTCPExTable = ^TMibTCPExTable;
  TMibTCPExTable = packed record
    dwNumEntries: DWORD;
    Table: array[0..0] of TMibTCPExRow;
  end;
  PTMibUdpExRow = ^TMibUdpExRow;
  TMibUdpExRow = packed record
    dwLocalAddr: DWORD;
    dwLocalPort: DWORD;
    dwProcessID: DWORD;
  end;
  PTMibUdpExTable = ^TMibUdpExTable;
  TMibUdpExTable = packed record
    dwNumEntries: DWORD;
    table: array [0..0] of TMibUdpExRow;
  end;
    //---------------------------------------------

  TProcessEntry32 = packed record
    dwSize: DWORD;
    cntUsage: DWORD;
    th32ProcessID: DWORD;
    th32DefaultHeapID: DWORD;
    th32ModuleID: DWORD;
    cntThreads: DWORD;
    th32ParentProcessID: DWORD;
    pcPriClassBase: Longint;
    dwFlags: DWORD;
    szExeFile: array [0..MAX_PATH - 1] of WideChar;
  end;
    //---------------------------------------------
 PTMibTCPRow = ^TMibTCPRow;
 TMibTCPRow = packed record
  dwState: DWORD;
  dwLocalAddr: DWORD;
  dwLocalPort: DWORD;
  dwRemoteAddr: DWORD;
  dwRemotePort: DWORD;
 end;


      //---------------------------------------------
```

```
PTMibTCPTable = ^TMibTCPTable;
TMibTCPTable = packed record
  dwNumEntries: DWORD;
  Table: array[0..0] of TMibTCPRow;
end;
    //---------------------------------------------
PTMibUdpRow = ^TMibUdpRow;
TMibUdpRow = packed record
  dwLocalAddr: DWORD;
  dwLocalPort: DWORD;
end;
    //---------------------------------------------
PTMibUdpTable = ^TMibUdpTable;
TMibUdpTable = packed record
  dwNumEntries: DWORD;
  table: array [0..0] of TMibUdpRow;
end;
    //---------------------------------------------
PGroupUsersInfo0 = ^_GROUP_USERS_INFO_0;
_GROUP_USERS_INFO_0 = packed record
  grui0_name: LPWSTR;
end;
TGroupUsersInfo0 = _GROUP_USERS_INFO_0;
GROUP_USERS_INFO_0 = _GROUP_USERS_INFO_0;
 _WKSTA_USER_INFO_1 = record
  wkui1_username: LPWSTR;
  wkui1_logon_domain: LPWSTR;
  wkui1_oth_domains: LPWSTR;
  wkui1_logon_server: LPWSTR;
end;
WKSTA_USER_INFO_1 = _WKSTA_USER_INFO_1;
PWKSTA_USER_INFO_1 = ^_WKSTA_USER_INFO_1;
LPWKSTA_USER_INFO_1 = ^_WKSTA_USER_INFO_1;
// MAC
TMacAddress = array[0..MAX_ADAPTER_ADDRESS_LENGTH] of byte;
    //---------------------------------------------
TMibIPNetRow = packed record
  dwIndex        : DWORD;
  dwPhysAddrLen   : DWORD;
  bPhysAddr       : TMACAddress;  // MAC!!!
  dwAddr         : DWORD;
  dwType         : DWORD;
end;
TMibIPNetRowArray = array [0..512] of TMibIPNetRow;
PTMibIPNetTable = ^TMibIPNetTable;
TMibIPNetTable = packed record
  dwNumEntries    : DWORD;
  Table: TMibIPNetRowArray;
end;
TDemoThread = class(TThread)
private
  TreeNetWrk: TTreeNode;
  TreeDomain: TTreeNode;
```

```
  TreeServer: TTreeNode;
  TreeShares: TTreeNode;
  Param_dwType: Byte;
  Param_dwDisplayType: Byte;
  Param_lpRemoteName: String;
  Param_lpIP: String;
protected
  procedure Execute; override;
  procedure Scan(Res: TNetResource; Root: boolean);
  procedure AddElement;
  procedure Stop;
end;
TIPEdit = class
private
  FHandle: THandle;
  FIP: Integer;
  FFont: Integer;
  function GetText: String;
  procedure SetText(const Value: String);
public
  constructor Create(AOwner: TWinControl; Rect: TRect);
  destructor Destroy; override;
  property Text: String read GetText write SetText;
end;
TForm1 = class(TForm)
    //---------------------------------------------
  …..
    //---------------------------------------------
  function GetNameFromIP(const IP: String): String;
  function GetUsers(const CompName: String): String;
  function GetDomain(const CompName, Provider: String): String;
  function GetComment(CompName, Provider: String): String;
  function GetProvider(const CompName: String): String;
  function GetMacFromIP(const IP: String): String;
  function GetDomainServer(const DomainName: String): String;
  function GetGroups(DomainServer: String; UserName: String): String;
  function GetShares(const CompName: String): String;
    //---------------------------------------------
  ….
    //---------------------------------------------
private
  Thread: TDemoThread;  IPFrom, IPTo: TIPEdit; FThreadCount, FCompFound: Integer;
  IP, Font: Integer; IPst:String;
  function PortStateToStr(const State: DWORD): String;
  procedure SetThreadCount(const Value: Integer);
  procedure SetCompFound(const Value: Integer);
  procedure GetMemoryInfo;
  procedure GetCompInfo;
  procedure GetIPInfo;
  procedure GetAdapterInfo;
  procedure UpdateDisk;
public
  property ThreadCount: Integer read FThreadCount write SetThreadCount;
```

```
  property CompFound: Integer read FCompFound write SetCompFound;
end;
LMSTR = LPWSTR;
NET_API_STATUS = DWORD;
PShareInfo1 = ^_SHARE_INFO_1;
_SHARE_INFO_1 = record
 shi1_netname: LMSTR;
 shi1_type: DWORD;
 shi1_remark: LMSTR;
end;
TShareInfo1 = _SHARE_INFO_1;
TScanThread = class(TThread)
private
  FIP: Integer;
  FRes: TStringList;
  function GetCompName(const Addr: Integer): String;
  procedure Scan;
  procedure UpdateTree;
  procedure IncCount;
  procedure DecCount;
protected
  procedure Execute; override;
public
  property IP: Integer read FIP write FIP;
end;
    //---------------------------------------------
    //---------------------------------------------
    //---------------------------------------------
{$EXTERNALSYM WNetGetResourceInformation}
function WNetGetResourceInformation(lpNetResource: PNetResource;
  lpBuffer: Pointer; var lpcbBuffer: DWORD; lplpSystem: Pointer): DWORD; stdcall;
{$EXTERNALSYM GetIpNetTable}
function GetIpNetTable(pIpNetTable: PTMibIPNetTable;
  pdwSize: PULONG; bOrder: Boolean): DWORD; stdcall;
function WNetGetResourceInformation; external mpr name 'WNetGetResourceInformationA';
function GetIpNetTable; external IPHLPAPI name 'GetIpNetTable';
function NetGetAnyDCName(servername: LPCWSTR;  domainname: LPCWSTR;
  bufptr: Pointer): Cardinal;
  stdcall; external 'netapi32.dll';
function NetShareEnum(servername: LMSTR; level: DWORD; var bufptr: Pointer;
 prefmaxlen: DWORD; entriesread, totalentries,
 resume_handle: LPDWORD): NET_API_STATUS; stdcall; external 'Netapi32.dll';
function NetApiBufferFree(Buffer: Pointer): NET_API_STATUS; stdcall; external 'Netapi32.dll';
function NetWkstaUserEnum(ServerName: LPCWSTR;
                    Level: DWORD;
                    BufPtr: Pointer;
                    PrefMaxLen: DWORD;
                    EntriesRead: LPDWORD;
                    TotalEntries: LPDWORD;
                    ResumeHandle: LPDWORD): LongInt; stdcall; external 'netapi32.dll';
function NetUserGetGroups(ServerName: LPCWSTR;
                    UserName: LPCWSTR;
                    level: DWORD;
```

```
                    bufptr: Pointer;
                    prefmaxlen: DWORD;
                    var entriesread: DWORD;
                    var totalentries: DWORD): LongInt; stdcall; external 'netapi32.dll';
  procedure InfComp;
  function GetTcpTable(pTCPTable: PTMibTCPTable; var pDWSize: DWORD;
    bOrder: BOOL): DWORD; stdcall; external 'IPHLPAPI.DLL';
  function GetUdpTable(pUDPTable: PTMibUDPTable; var pDWSize: DWORD;
    bOrder: BOOL): DWORD; stdcall; external 'IPHLPAPI.DLL';
    function AllocateAndGetTcpExTableFromStack(pTCPExTable: PTMibTCPExTable;
      bOrder: BOOL; heap: THandle; zero: DWORD; flags: DWORD): DWORD; stdcall;
      external 'IPHLPAPI.DLL';
    function AllocateAndGetUdpExTableFromStack(pUDPExTable: PTMibUDPExTable;
      bOrder: BOOL; heap: THandle; zero: DWORD; flags: DWORD): DWORD; stdcall;
      external 'IPHLPAPI.DLL';
    function CreateToolhelp32Snapshot(dwFlags, th32ProcessID: DWORD): THandle;
      stdcall; external 'KERNEL32.DLL';
    function Process32First(hSnapshot: THandle; var lppe: TProcessEntry32): BOOL;
      stdcall; external 'KERNEL32.DLL' name 'Process32FirstW';
    function Process32Next(hSnapshot: THandle; var lppe: TProcessEntry32): BOOL;
      stdcall; external 'KERNEL32.DLL' name 'Process32NextW';
type
  TRes = record
    S: Extended;  Q: Extended;  end;
const
  N = 10;
var
  Form1: TForm1;
  Y, O, Z, E_, F1, F2, Res_S, Res_Q, B: array[1..255] of Extended;
  F, C, E, OO: array[1..255] of array[1..255] of Extended;
  arOp:array [1..7,1..4]of string;
implementation
{$R *.dfm}
function RoundFloat(R: Extended; Decimals: Integer): Extended;
var
  Factor: Extended;
Begin   Factor := Int(Exp(Decimals * Ln(10)));  Result := Round(Factor * R) / Factor; end;
function M(j: Integer): TRes;
var
  i, i1, i2: Integer;
  S1, S2, Q1, Q2: Extended;
begin
      //---------------------------------------------
  S1 := 0;
  for i := 1 to N do
    S1 := S1 + O[i]*F[i,j];
  for i1 := 1 to N do
    for i2 := 1 to N do
      if (i2 <> i1) then
        S1 := S1 + OO[i1,i2]*F[i1,j]*F[i2,j];
  S1 := abs(S1 - Y[j]);
  //ShowMessage(Floattostr(S1));
  S2 := 0;
```

```
  for i := 1 to N do
    S2 := S2 + O[i]*C[i,j];
  for i1 := 1 to N do
    for i2 := 1 to N do
      if (i2 <> i1) then
        S2 := S2 + OO[i1,i2]*(F[i1,j]*C[i2,j] + F[i2,j]*C[i1,j]);
  M.S := S1/S2;
  Q1 := 0;
  for i := 1 to N do
    Q1 := Q1 + O[i]*F[i,j];
  for i1 := 1 to N do
    for i2 := 1 to N do
      if (i2 <> i1) then
        Q1 := Q1 + OO[i1,i2]*F[i1,j]*F[i2,j];
  Q1 := abs(Y[j] - Q1);
  Q2 := 0;
  for i := 1 to N do
    Q2 := Q2 + O[i]*E[i,j];
  for i1 := 1 to N do
    for i2 := 1 to N do
      if (i2 <> i1) then
        Q2 := Q2 + OO[i1,i2]*(F[i1,j]*E[i2,j] + F[i2,j]*E[i1,j]);
  M.Q := Q1/Q2;
end;

function Replace(Str, X, Y: string): string;
var
  buf1, buf2, buffer: string;
  i: Integer;
begin
  buf1 := '';
  buf2 := Str;
  Buffer := Str;
  while Pos(X, buf2) > 0 do
  begin
    buf2 := Copy(buf2, Pos(X, buf2), (Length(buf2) - Pos(X, buf2)) + 1);
    buf1 := Copy(Buffer, 1, Length(Buffer) - Length(buf2)) + Y;
    Delete(buf2, Pos(X, buf2), Length(X));
    Buffer := buf1 + buf2;
  end;
  Replace := Buffer;
end;
procedure TForm1.SetCompFound(const Value: Integer);
begin
  FCompFound := Value;
  StatusBar1.Panels.Items[1].Text := Format(RES_COMPCOUNT, [Value]);
  Application.ProcessMessages;
end;
procedure TForm1.SetThreadCount(const Value: Integer);
begin
  if Value < FThreadCount then
    ProgressBar.Position := ProgressBar.Max - Value;
  FThreadCount := Value;
```

```
  StatusBar1.Panels.Items[0].Text := Format(RES_THREADCOUNT, [Value]);
  if Value = 0 then
  begin
    ProgressBar.Position := 0;
    btnStart.Enabled := True;
  end;
  Application.ProcessMessages;
end;
procedure ScanLocalNW();
begin
  with form1 do begin
    Tag := Tag + 1;
    if (Tag mod 2) = 1 then begin
      TreeView1.Items.Clear;
      StatusBar1.Panels[1].Text := STR_STARTED;
      Thread := TDemoThread.Create(False);
    end
    else  begin
      StatusBar1.Panels[1].Text := STR_END;
      Thread.Terminate;
    end;   end;   end;
function GetIPAddress(NetworkName: String): String;
var
 Error: DWORD;
 HostEntry: PHostEnt;
 Data: WSAData;
 Address: In_Addr;
begin
  Delete(NetworkName, 1, 2);
  Error:=WSAStartup(MakeWord(1, 1), Data);
  if Error = 0 then
  begin
    HostEntry:=gethostbyname(PChar(NetworkName));
    Error:=GetLastError;
    if Error = 0 then
    begin
      Address:=PInAddr(HostEntry^.h_addr_list^)^;
      Result:=inet_ntoa(Address);
    End  else
     Result:='Unknown';
  End   else
    Result:='Error';
  WSACleanup;
end;
{ TDemoThread }
procedure TDemoThread.Execute;
var
  R:TNetResource;
begin
  inherited;
  Priority := tpIdle;
  FreeOnTerminate := True;
  Resume;
```

```
  Scan(R, True);
  TreeDomain := nil;
  TreeServer := nil;
  Synchronize(Stop);
end;
procedure TDemoThread.Scan(Res: TNetResource; Root: boolean);
var
 hEnum: Cardinal;
 nrResource: array[0..512] of TNetResource;
 dwSize: DWORD;
 numEntries: DWORD;
 I: DWORD;
 dwResult: DWORD;
begin
  if Root then
    dwResult := WNetOpenEnum(RESOURCE_GLOBALNET, RESOURCETYPE_ANY,
      0, nil, hEnum)
  else
    dwResult := WNetOpenEnum(RESOURCE_GLOBALNET, RESOURCETYPE_ANY,
      0, @Res, hEnum);
  if dwResult = NO_ERROR then
  begin
    dwSize := SizeOf(nrResource);
    numEntries := DWORD(-1);                          // ERROR_NO_MORE_ITEMS
    if WNetEnumResource(hEnum, numEntries, @nrResource, dwSize) = NO_ERROR then
    begin       for i := 0 to numEntries - 1 do
      begin         if Terminated then Break;
        with nrResource[i] do
        begin
          Param_dwType := dwType;
          Param_dwDisplayType := dwDisplayType;
          Param_lpRemoteName := lpRemoteName;
          if Param_dwDisplayType = RESOURCEDISPLAYTYPE_SERVER then
            Param_lpIP := GetIPAddress(Param_lpRemoteName);
        end;
        if Assigned(nrResource[i].lpRemoteName) then
          Synchronize(AddElement);
        Scan(nrResource[i], false);
      end;
    WNetCloseEnum(hEnum);
    end;   end; end;
procedure TDemoThread.AddElement;
begin
  Application.ProcessMessages;
  case Param_dwDisplayType of
    RESOURCEDISPLAYTYPE_NETWORK:
    begin
      TreeNetWrk := Form1.TreeView1.Items.Add(nil, Param_lpRemoteName);
      TreeNetWrk.StateIndex := 1;
    end;
    RESOURCEDISPLAYTYPE_DOMAIN:
    begin
      TreeDomain := Form1.TreeView1.Items.AddChild(TreeNetWrk, Param_lpRemoteName);
```

```
      TreeDomain.StateIndex := 2;
    end;
    RESOURCEDISPLAYTYPE_SERVER:
    begin
      TreeServer := Form1.TreeView1.Items.AddChild(TreeDomain, Param_lpRemoteName + ' IP: '
+ Param_lpIP);
      TreeServer.StateIndex := 3;
    end;
    RESOURCEDISPLAYTYPE_SHARE:
    begin
      TreeShares := Form1.TreeView1.Items.AddChild(TreeServer, Param_lpRemoteName);
      TreeShares.StateIndex := 3 + Param_dwType;
    end;   end;  end;
procedure TDemoThread.Stop;
begin
  Form1.StatusBar1.Panels[1].Text := STR_STOPPED;
  Form1.Tag := 0;
end;
{ TForm1 }
procedure ReadFile();
var
   i,j,v_pos1,v_pos2:Integer;
   str1,str2:string;
   f : TStringList;
begin
  f := TStringList.Create();
  f.LoadFromFile('tab.dbf');
  for i := 0 to f.Count-1 do begin
    v_pos1:=1;
    str1:=f[i];
    for j:=0 to 3 do begin
      v_pos2:=pos(' ',str1);
      str2:=copy(str1,1,v_pos2-1);
      form1.AdvStringGrid2.Cells[j+1,i+1]:=str2;
      delete(str1,1,v_pos2);
      v_pos1:=v_pos2;
    end;   end;
  f.Free;
  f := TStringList.Create();
  f.LoadFromFile('tab2.dbf');
  v_pos1:=1;
  str1:=f[0];
  for j:=0 to 3 do begin
     v_pos2:=pos(' ',str1);
     str2:=copy(str1,1,v_pos2-1);
     form1.AdvStringGrid1.Cells[1,j]:=str2;
     delete(str1,1,v_pos2);
     v_pos1:=v_pos2;
  end;
  f.Free;
  f := TStringList.Create();
  f.LoadFromFile('tab3.dbf');
  v_pos1:=1;
```

```
  str1:=f[0];
  for j:=0 to 6 do begin
    v_pos2:=pos(' ',str1);
    str2:=copy(str1,1,v_pos2-1);
    form1.AdvStringGrid3.Cells[1,j]:=str2;
    delete(str1,1,v_pos2);
    v_pos1:=v_pos2;
  end;
  f.Free;
end;
procedure WriteFile();
var
  f: textFile      //-------------------------------------------

  str: string;       //-------------------------------------------

  i,j:Integer;
begin
   AssignFile(f,'tab.dbf');
   Rewrite(f);
   for i := 0 to 6 do begin      str:='';
    for j:=0 to 3 do str:=str+form1.AdvStringGrid2.Cells[j+1,i+1]+' ';
    Writeln(f, str);
   end;
   Closefile(f);
   AssignFile(f,'tab2.dbf');
   Rewrite(f);
   str:='';
   for j:=0 to 3 do str:=str+form1.AdvStringGrid1.Cells[1,j]+' ';
   Writeln(f, str);
   Closefile(f);
   AssignFile(f,'tab3.dbf');
   Rewrite(f);
   str:='';
   for j:=0 to 6 do str:=str+form1.AdvStringGrid3.Cells[1,j]+' ';
   Writeln(f, str);
   Closefile(f);
end;
procedure TForm1.FormCreate(Sender: TObject);
var
  i,j:byte;
begin
  application.Title:='Аналізатор уразливостей';
  Application.HintHidePause:=40000;
  Tag := 0;
  IPFrom := TIPEdit.Create(gbAddrRange, Rect(32, 16, 121, 21));
  IPFrom.Text := '192.168.1.4';
  IPTo := TIPEdit.Create(gbAddrRange, Rect(32, 40, 121, 21));
  IPTo.Text := '192.168.1.10';
     //-------------------------------------------
  IP := MAKEIPADDRESS(192, 168, 1, 4);
  ScanLocalNW;
AdvStringGrid2.Cells[0,1]:='1. Визначення інформації, що підлягає захисту';
```

```
  AdvStringGrid2.Cells[0,2]:='2. Виявлення загроз та каналів витоку інформації';
  AdvStringGrid2.Cells[0,3]:='3. Проведення оцінки вразливості та ризиків';
  AdvStringGrid2.Cells[0,4]:='4. Визначення вимог до СЗІ;
  AdvStringGrid2.Cells[0,5]:='5. Здійснення вибору засобів захисту;
  AdvStringGrid2.Cells[0,6]:='6. Впровадження та використання обраних заходів та засобів';
  AdvStringGrid2.Cells[0,7]:='7. Контроль цілісності та управління захистом';
  AdvStringGrid3.Cells[0,0]:='1. Визначення інформації, що підлягає захисту';
  AdvStringGrid3.Cells[0,1]:='2. Виявлення загроз та каналів витоку інформації';
  AdvStringGrid3.Cells[0,2]:='3. Проведення оцінки вразливості та ризиків';
  AdvStringGrid3.Cells[0,3]:='4. Визначення вимог до СЗІ;
  AdvStringGrid3.Cells[0,4]:='5. Здійснення вибору засобів захисту;
  AdvStringGrid3.Cells[0,5]:='6. Впровадження та використання обраних заходів та засобів';
  AdvStringGrid3.Cells[0,6]:='7. Контроль цілісності та управління захистом';
AdvStringGrid2.Cells[1,0]:='Рівень  1';
 AdvStringGrid2.Cells[2,0]:='Рівень  2';
 AdvStringGrid2.Cells[3,0]:='Рівень  3';
 AdvStringGrid2.Cells[4,0]:='Рівень  4';
 AdvStringGrid4.Cells[0,0]:='Рівень  1';
 AdvStringGrid4.Cells[1,0]:='Рівень  2';
 AdvStringGrid4.Cells[2,0]:='Рівень  3';
 AdvStringGrid4.Cells[3,0]:='Рівень  4';
 AdvStringGrid1.Cells[0,0]:='Рівень  1';
 AdvStringGrid1.Cells[0,1]:='Рівень  2';
 AdvStringGrid1.Cells[0,2]:='Рівень  3';
 AdvStringGrid1.Cells[0,3]:='Рівень  4';
AdvStringGrid8.Cells[0,0] := 'Діапазон значень';
  AdvStringGrid8.Cells[0,1] := 'у %';
  AdvStringGrid8.Cells[0,2] := 'ні(0) та(1)';
  AdvStringGrid8.Cells[0,3] := 'Мбіт';
  AdvStringGrid8.Cells[0,4] := 'ні(0) та(1)';
  AdvStringGrid8.Cells[0,5] := 'Мбіт';
  AdvStringGrid8.Cells[0,6] := 'кількість...';
  AdvStringGrid8.Cells[0,7] := 'ні(0) та(1)';
  AdvStringGrid8.Cells[0,8] := 'у %';
  AdvStringGrid8.Cells[0,9] := 'ні(0) та(1)';
  AdvStringGrid8.Cells[0,10] := 'у біт/с';
  AdvStringGrid9.Cells[0,0] := 'Результат обчислень';
  AdvStringGrid9.Cells[1,0] := 'S';
  AdvStringGrid9.Cells[2,0] := 'Q';
  AdvStringGrid7.Cells[0,0] := 'Критерії';
  AdvStringGrid7.Cells[0,1] := 'Надлишковий трафік';
  AdvStringGrid7.Cells[0,2] := 'Наявність підозрілих процесів';
  AdvStringGrid7.Cells[0,3] := 'Великі пакети або фрагменти';
  AdvStringGrid7.Cells[0,4] := 'Невідповідність IP адрес в ICMP-пакетах';
  AdvStringGrid7.Cells[0,5] := 'Розмір ICMP-пакетів';
  AdvStringGrid7.Cells[0,6] := 'Відкриті порти';
  AdvStringGrid7.Cells[0,7] := 'Включені опції в TCP/UDP-пакетах';
  AdvStringGrid7.Cells[0,8] := 'Велика кількість pingзапитів/відповідей';
  AdvStringGrid7.Cells[0,9] := 'Відсутність ping-а';
  AdvStringGrid7.Cells[0,10] := 'Уповільнення швидкості з'єднання';
  AdvStringGrid9.Cells[0,0] := 'За критеріями';
  AdvStringGrid9.Cells[0,1] := 'По трафік';
  AdvStringGrid9.Cells[0,2] := 'За наявності підозрілих процесів';
```

```
 AdvStringGrid9.Cells[0,3] := 'За пакетами або фрагментами';
 AdvStringGrid9.Cells[0,4] := 'За невідповідністю IP адрес в ICMP-пакетах';
 AdvStringGrid9.Cells[0,5] := 'За розміром ICMP-пакетів';
 AdvStringGrid9.Cells[0,6] := 'По відкритих портах';
 AdvStringGrid9.Cells[0,7] := 'За включенням опції в TCP/UDP-пакетах';
 AdvStringGrid9.Cells[0,8] := 'За ping запитами/відповідями';
 AdvStringGrid9.Cells[0,9] := 'За відсутністю ping-а';
 AdvStringGrid9.Cells[0,10] := 'З уповільнення швидкості з'єднання';
 AdvStringGrid7.Cells[1,0] := 'Оцінка';
 AdvStringGrid7.Cells[1,1] := '5';
 AdvStringGrid7.Cells[1,2] := '3';
 AdvStringGrid7.Cells[1,3] := '7';
 AdvStringGrid7.Cells[1,4] := '4';
 AdvStringGrid7.Cells[1,5] := '8';
 AdvStringGrid7.Cells[1,6] := '4';
 AdvStringGrid7.Cells[1,7] := '5';
 AdvStringGrid7.Cells[1,8] := '6';
 AdvStringGrid7.Cells[1,9] := '9';
 AdvStringGrid7.Cells[1,10] := '3';
for i := 1 to N do
  AdvStringGrid9.Cells[0,j] := Floattostr(i);
for i := 1 to N do
  AdvStringGrid8.Cells[i,0] := Floattostr(i);
for i := 1 to N do begin
  F[i,1] := i*10;
  F[i,2] := 1;
  F[i,3] := 96;
  F[i,4] := 1;
  F[i,5] := i*10;
  F[i,6] := i;
  F[i,7] := 1;
  F[i,8] := i*10;
  F[i,9] := 1;
  F[i,10] := i*10;
end;
for i:=1 to N do     for j:=1 to N do
    C[i,j]:= (i / j) / N + 0.1 ;   for i:=1 to N do
  for j:=1 to N do begin       E[i,j]:= (i / j) / N + 0.4;
    E_[i]:= (i / j) / N + 0.4;   end;
for i:=1 to N do begin
  F1[i]:=F[10,i];
  F2[i]:=F[1,i];
end;
for i:=1 to N do
  Z[i] := abs(F1[i]-F2[i]/E_[i]);
for i := 1 to N do
    Y[i]:= O[i]+Z[i]+O[i]*F1[i];
for i:=1 to N do
  for j:=1 to N do
    AdvStringGrid8.Cells[i,j]:=Floattostr(F[i,j]);
ReadFile();
 for i := 1 to 7 do
   for j:=1 to 4 do form1.AdvStringGrid4.Cells[j-1,i]:='0';
```

arOp[1,1]:='Виклад у законодавчих, нормативних та методичних документах питань, що визначають перелік відомостей, що використовуються в процесах та програмах ІС, які підлягають захисту та порядок визначення таких відомостей.';
arOp[1,2]:='Опис функцій органів, відповідальних за визначення відомостей, що підлягають захисту при використанні їх у процесах та програмах ІС.';
arOp[1,3]:='Опис заходів (політики безпеки), що забезпечують своєчасне та якісне визначення переліку відомостей, що підлягають захисту при використанні їх у процесах та програмах ІС.';
arOp[1,4]:='Опис набору засобів для забезпечення оперативності та якості визначення інформації, що підлягає захисту при використанні їх у процесах та програмах ІС.';
arOp[2,1]:='Виклад у законодавчих, нормативних та методичних документах питань, що визначають порядок виявлення потенційних каналів витоку інформації в процесах та програмах ІС.';
arOp[2,2]:='Опис функцій органів, відповідальних за виявлення потенційних каналів витоку інформації в процесах та програмах ІС.';
arOp[2,3]:='Опис заходів (політики безпеки), що забезпечують своєчасне та якісне виявлення потенційних каналів витоку інформації в процесах та програмах ІС.';
arOp[2,4]:='Опис набору засобів для забезпечення оперативності та якості виявлення потенційних каналів витоку інформації в процесах та програмах ІС.';
arOp[3,1]:='Виклад у законодавчих, нормативних та методичних документах питань, що визначають проведення оцінки вразливості та ризиків для інформації в процесах та програмах ІС.';
arOp[3,2]:='Опис функцій органів, відповідальних за проведення оцінки вразливості та ризиків для інформації в процесах та програмах ІС.';
arOp[3,3]:='Опис заходів (політики безпеки), що забезпечують своєчасне та якісне проведення оцінки вразливості та ризиків для інформації в процесах та програмах ІС.';
arOp[3,4]:='Опис набору коштів для забезпечення оперативності та якості проведення оцінки вразливості та ризиків для інформації в процесах та програмах ІС.';
arOp[4,1]:='Виклад у законодавчих, нормативних та методичних документах питань визначення вимог до СЗІ у процесах та програмах ІС.';
arOp[4,2]:='Опис функцій органів, відповідальних за визначення вимог до СЗІ у процесах та програмах ІС.';
arOp[4,3]:='Опис заходів (політики безпеки), що забезпечують своєчасне та якісне визначення вимог до СЗІ у процесах та програмах ІС.';
arOp[4,4]:='Опис набору коштів для забезпечення оперативності та якості визначення вимог до СЗІ у процесах та програмах ІС.';
arOp[5,1]:='Виклад у законодавчих, нормативних та методичних документах питань, що визначають здійснення вибору засобів захисту для інформації в процесах та програмах ІС.';
arOp[5,2]:='Опис функцій органів, відповідальних за здійснення вибору засобів захисту для інформації в процесах та програмах ІС.';
arOp[5,3]:='Опис заходів (політики безпеки), що забезпечують своєчасне та якісне здійснення вибору засобів захисту для інформації в процесах та програмах ІС.';
arOp[5,4]:='Опис набору засобів для забезпечення оперативності та якості здійснення вибору засобів захисту для інформації в процесах та програмах ІС.';
arOp[6,1]:='Виклад у законодавчих, нормативних та методичних документах питань, що визначають порядок впровадження та використання обраних заходів та засобів захисту для інформації в процесах та програмах ІС.';
arOp[6,2]:='Опис функцій органів, відповідальних за впровадження та використання обраних заходів та засобів захисту для інформації в процесах та програмах ІС.';
arOp[6,3]:='Опис заходів (політики безпеки), що забезпечують своєчасне та якісне впровадження та використання обраних способів та засобів захисту для інформації в процесах та програмах ІС.';

```
  arOp[6,4]:='Опис набору засобів для забезпечення оперативності та якості впровадження та використання обраних заходів та засобів захисту для інформації в процесах та програмах ІС.';
  arOp[7,1]:='Виклад у законодавчих, нормативних та методичних документах питань, що визначають порядок контролю цілісності та управління захистом для інформації в процесах та програмах ІС.';
  arOp[7,2]:='Опис функцій органів, відповідальних за здійснення контролю цілісності та управління захистом для інформації в процесах та програмах ІС.';
  arOp[7,3]:='Опис заходів (політики безпеки), що забезпечують своєчасний та якісний контроль цілісності та управління захистом для інформації в процесах та програмах ІС.';
  arOp[7,4]:='Опис набору засобів для забезпечення оперативності та якості контролю цілісності та управління захистом для інформації в процесах та програмах ІС.';end;
procedure TForm1.AdvStringGrid7KeyPress(Sender: TObject; var Key: Char);
begin
 if not (key in ['0'..'9','.']) then
 key := #0;
end;
procedure TForm1.AdvStringGrid8KeyPress(Sender: TObject; var Key: Char);
begin
 if not (key in ['0'..'9','.']) then
 key := #0;
end;
procedure TForm1.Button1Click(Sender: TObject);
begin
 ScanLocalNW;
end;
procedure TForm1.TreeView1Click(Sender: TObject);
var   p1,a,b,c,d,p2:integer;
 st:string;
begin
 if Assigned(TreeView1.Selected) then begin
   StatusBar1.Panels[0].Text := '   ' + TreeView1.Selected.Text;
   st:=TreeView1.Selected.Text;
   p1:=Pos(':',st);
   if p1<>0 then begin
     IPst:=Copy(st,p1+1,length(st)-p1);
     p2:=Pos('.',st);
     a:=StrToInt(Copy(st, p1+1, p2-p1-1));
     st[p2]:=',';
     p1:=p2;
     p2:=Pos('.',st);
     b:=StrToInt(Copy(st, p1+1, p2-p1-1));
     st[p2]:=',';
     p1:=p2;
     p2:=Pos('.',st);
     c:=StrToInt(Copy(st, p1+1, p2-p1-1));
     st[p2]:=',';
     p1:=p2;
     p2:=length(st)-p1;
     d:=StrToInt(Copy(st, p1+1, p2));
     IP := MAKEIPADDRESS(a,b,c,d);
     InfComp;
   end;
```

```
  end
  else StatusBar1.Panels[0].Text := STR_FIELD;
end;
procedure TForm1.TreeView1DblClick(Sender: TObject);
var
  Str: String;
begin
  if Assigned(TreeView1.Selected) then
  begin
    Str := TreeView1.Selected.Text;
    if Copy(Str, 1, 2) <> '\\' then Exit;
    if Pos(' IP:', Str) <> 0 then
      ShellExecute(Handle, 'explore', PChar(Copy(Str, 1, Pos(' IP:', Str))), nil, nil, SW_SHOW)
    Else       ShellExecute(Handle, 'explore', PChar(Str), nil, nil, SW_SHOW);
  end; end;

procedure TForm1.btnStartClick(Sender: TObject);
var
  I, AFrom, ATo: Integer;
  Prefix: String;
  function ValidRange: Boolean;
  var     F, T: TInAddr;
  begin
    F.S_addr := inet_addr(PChar(IPFrom.Text));
    T.S_addr := inet_addr(PChar(IPTo.Text));
    Result := (F.S_un_b.s_b1 = T.S_un_b.s_b1) and
              (F.S_un_b.s_b2 = T.S_un_b.s_b2) and
              (F.S_un_b.s_b3 = T.S_un_b.s_b3);
    if Result then
    begin
      AFrom := Integer(F.S_un_b.s_b4);
      ATo := Integer(T.S_un_b.s_b4);
      Prefix := IntToStr(Integer(F.S_un_b.s_b1)) + '.' +
                IntToStr(Integer(F.S_un_b.s_b2)) + '.' +
                IntToStr(Integer(F.S_un_b.s_b3)) + '.';
      ProgressBar.Max := ATo - AFrom;
      ProgressBar.Position := 0;
    end
    else
      MessageDlg(RES_ERR_RANGE, mtError, [mbOK], 0);
  end;
begin
  CompFound := 0;
  ThreadCount := 0;
  tvResult.Items.Clear;
  if ValidRange then
  begin
    btnStart.Enabled := False;
    for I := AFrom to ATo do
      with TScanThread.Create(False) do
      begin
        IP := inet_addr(PChar(Prefix + IntToStr(I)));
        FreeOnTerminate := True;
```

```
      Resume;
    end;   end; end;

constructor TIPEdit.Create(AOwner: TWinControl; Rect: TRect);
begin
  InitCommonControl(ICC_INTERNET_CLASSES);
  FHandle:= CreateWindow(WC_IPADDRESS, nil, WS_CHILD or WS_VISIBLE,
    Rect.Left, Rect.Top, Rect.Right, Rect.Bottom, AOwner.Handle, 0, hInstance, nil);
  FFont := CreateFont(-11, 0, 0, 0, 400, 0, 0, 0, DEFAULT_CHARSET,
    OUT_DEFAULT_PRECIS, CLIP_DEFAULT_PRECIS, DEFAULT_QUALITY,
    DEFAULT_PITCH or FF_DONTCARE, 'MS Sans Serif');
  SendMessage(FHandle, WM_SETFONT, FFont, 0);
  Text := '0.0.0.0';
end;

destructor TIPEdit.Destroy;
begin   DeleteObject(FFont);
  inherited;
end;
function TIPEdit.GetText: String;
begin
  SendMessage(FHandle, IPM_GETADDRESS, 0, Longint(PDWORD(@FIP)));
  Result := IntToStr(FIRST_IPADDRESS(FIP))+
     '.' + IntToStr(SECOND_IPADDRESS(FIP)) +
     '.' + IntToStr(THIRD_IPADDRESS(FIP)) +
     '.' + IntToStr(FOURTH_IPADDRESS(FIP));
end;
procedure TIPEdit.SetText(const Value: String);
  function MakeIPAddressEx(b1, b2, b3, b4: Char):LPARAM;
  begin
    Result := MAKEIPADDRESS(DWORD(b1), DWORD(b2), DWORD(b3), DWORD(b4));
  end;
var
  Tmp: TInAddr;
begin
  Tmp.S_addr := inet_addr(PChar(Value));
  if Tmp.S_addr = INADDR_NONE then Exit;
  with Tmp.S_un_b do
    FIP := MakeIPAddressEx(s_b1, s_b2, s_b3, s_b4);
  SendMessage(FHandle, IPM_SETADDRESS, 0, FIP);
end;
{TScanThread }
procedure TScanThread.DecCount;
begin
  Form1.ThreadCount := Form1.ThreadCount - 1;
end;
procedure TScanThread.Execute;
begin
  inherited;
  Synchronize(IncCount);
  Scan;
  Synchronize(DecCount);
end;
```

```
function TScanThread.GetCompName(const Addr: Integer): String;
var
  WSA: TWSAData;
  Host: PHostEnt;
  Err: Integer;
begin
  Result := RES_UNKNOWN;
  Err := WSAStartup(WSA_TYPE, WSA);
  if Err <> 0 then  //
  begin            //

ShowMessage(SysErrorMessage(GetLastError));

    Exit;
  end;
  try
    if Addr = INADDR_NONE then Exit;
    Host := gethostbyaddr(@Addr, SizeOf(Addr), PF_INET);
    if Assigned(Host) then
      Result := Host.h_name     else
  finally
    WSACleanup;
  end; end;
procedure TScanThread.IncCount;
begin
  Form1.ThreadCount := Form1.ThreadCount + 1;
end;
procedure TScanThread.Scan;
type
  TShareInfo1Array = array of TShareInfo1;
var
  entriesread, totalentries: DWORD;
  Info: Pointer;
  I: Integer;
  CompName: PWideChar;
begin
  CompName := StringToOleStr(GetCompName(FIP));
  if CompName = RES_UNKNOWN then Exit;
  FRes := TStringList.Create;
  try
    Fres.Add(CompName);
    if NetShareEnum(CompName, 1, Info, DWORD(-1), @entriesread,
      @totalentries, nil) = 0 then
    try
     if entriesread > 0 then
     begin
       for I := 0 to entriesread - 1 do
         FRes.Add(TShareInfo1Array(@(Info^))[I].shi1_netname);
       Synchronize(UpdateTree);
     end;
    finally
     NetApiBufferFree(Info);
    end;
```

```
  finally
    FRes.Free;
  end; end;

procedure TScanThread.UpdateTree;
var
  I: Integer;
  Root: TTreeNode;
begin
  Form1.tvResult.Items.BeginUpdate;
  try
    Root := Form1.tvResult.Items.Add(nil, FRes.Strings[0]);
    for I := 1 to FRes.Count - 1 do
      Form1.tvResult.Items.AddChild(Root, FRes.Strings[I]);
      Form1.CompFound := Form1.CompFound + 1;
  finally
    Form1.tvResult.Items.EndUpdate;
  end; end;
procedure InfComp;
var
  TmpCompName, TmpProvider, TmpGroup, TmpUser, TmpServer: String;
  Time: Cardinal;
  IPStr: String;
begin
 with Form1 do begin
    Time := GetTickCount;  //
       IPStr := IntToStr(FIRST_IPADDRESS(IP));
    IPStr := IPStr + '.' + IntToStr(SECOND_IPADDRESS(IP));
    IPStr := IPStr + '.' + IntToStr(THIRD_IPADDRESS(IP));
    IPStr := IPStr + '.' + IntToStr(FOURTH_IPADDRESS(IP));
    with memInfo, memInfo.Lines do begin
      Clear;                                    //
      Refresh;                                    //
                                              //
      Add(RES_IP + IPStr);                              // IP адреса
      TmpCompName := GetNameFromIP(IPStr);
      if TmpCompName = RES_UNKNOWN then Exit;
      Add(RES_CMP + TmpCompName);                           //
      TmpUser := GetUsers(IPStr);
      Add(RES_USR + TmpUser);                             //
      TmpProvider := GetProvider(TmpCompName);
      Add(RES_PROV + TmpProvider);                          //
      Add(RES_COM + GetComment(TmpCompName,
        TmpProvider));                                  // коментар
      TmpGroup := GetDomain(TmpCompName, TmpProvider);
      Add(RES_DOM + TmpGroup);                              // група
      TmpServer := GetDomainServer(TmpGroup);
      if TmpServer <> '' then begin
        Add(RES_SER + TmpServer);                           // Сервер
        Add(RES_GRP + GetGroups(TmpServer, TmpUser));
      end;
      Add(RES_SHARES + GetShares(TmpCompName));
      Add(RES_MAC + GetMacFromIP(IPStr));
```

```
      Add(RES_TIME + IntToStr(GetTickCount - Time));
    end;
end;
end;

function TForm1.GetNameFromIP(const IP: String): String;
var
  WSA: TWSAData;
  Host: PHostEnt;
  Addr: Integer;
  Err: Integer;
begin
  Result := RES_UNKNOWN;
  Err := WSAStartup(WSA_TYPE, WSA);
  if Err <> 0 then  //
  begin           //
    //ShowMessage(SysErrorMessage(GetLastError));
    Exit;
  end;
  try
    Addr := inet_addr(PChar(IP));
    if Addr = INADDR_NONE then
    begin
      //ShowMessage(SysErrorMessage(GetLastError));
      WSACleanup;
      Exit;
    end;
    Host := gethostbyaddr(@Addr, SizeOf(Addr), PF_INET);
    if Assigned(Host) then  //
      Result := Host.h_name //
    else
      //ShowMessage(SysErrorMessage(GetLastError));
  finally
    WSACleanup;
  end;

end;

function TForm1.GetUsers(const CompName: String): String;
var
  Buffer, tmpBuffer: Pointer;
  PrefMaxLen       : DWORD;
  Resume_Handle    : DWORD;
  EntriesRead      : DWORD;
  TotalEntries     : DWORD;
  I, Size          : Integer;
  PSrvr            : PWideChar;
begin
  PSrvr := nil;
  try
    // PWideChar
    Size := Length(CompName);
    GetMem(PSrvr, Size * SizeOf(WideChar) + 1);
```

```
    StringToWideChar(CompName, PSrvr, Size + 1);
    PrefMaxLen := DWORD(-1);
    EntriesRead := 0;
    TotalEntries := 0;
    Resume_Handle := 0;
    Buffer := nil;
    //
    if NetWkstaUserEnum( PSrvr, 1, @Buffer, PrefMaxLen, @EntriesRead,
      @TotalEntries, @Resume_Handle) = S_OK then
    begin
      tmpBuffer := Pointer(DWORD(Buffer) + SizeOf(WKSTA_USER_INFO_1));
      for I := 1 to TotalEntries - 1 do
      begin
        Result := Result + WKSTA_USER_INFO_1(tmpBuffer^).wkui1_username + ', ';
        tmpBuffer := Pointer(DWORD(tmpBuffer) + SizeOf(WKSTA_USER_INFO_1));
      end;
      Result := Copy(Result, 1, Length(Result) - 2);
  end
  else
ShowMessage(SysErrorMessage(GetLastError));
  finally
    NetApiBufferFree(Buffer);
    FreeMem(PSrvr);
  end; end;
function TForm1.GetDomain(const CompName, Provider: String): String;
var
  CurrRes: TNetResource;
  ParentName: array [0..1] of TNetResource;
  Enum: DWORD;
  Err: Integer;
begin
  with CurrRes do
  begin
    dwScope := RESOURCE_GLOBALNET;
    dwType := RESOURCETYPE_DISK;
    dwDisplayType := RESOURCEDISPLAYTYPE_SERVER;
    dwUsage := RESOURCEUSAGE_CONTAINER;
    lpLocalName := '';
    lpRemoteName := PChar('\\' + CompName);
    lpComment := '';
    lpProvider := PChar(Provider);
  end;
  Enum := SizeOf(ParentName);
  Err := WNetGetResourceParent(@CurrRes, @ParentName, Enum);
  if Err = NO_ERROR then
  begin
    Result := ParentName[0].lpRemoteName;
    if Result = '' then Result := RES_COM_NO;
  end
  else
    //ShowMessage(SysErrorMessage(GetLastError));
end;
function TForm1.GetComment(CompName, Provider: String): String;
```

```
var
  StopScan: Boolean;
  TmpRes: TNetResource;
  // Сканирование
  procedure Scan(Res: TNetResource; Root: boolean);
  var
    Enum, I: Cardinal;
    ScanRes: array [0..512] of TNetResource; //
    Size, Entries, Err: DWORD;              //

  begin
     if StopScan then Exit; //
    if Root = True then
     Err := WNetOpenEnum(RESOURCE_GLOBALNET, RESOURCETYPE_DISK,
       0, nil, Enum) //...
    else
     Err := WNetOpenEnum(RESOURCE_GLOBALNET, RESOURCETYPE_DISK,
       0, @Res, Enum); //
    if Err = NO_ERROR then
    begin
     Size := SizeOf(ScanRes);
     Entries := DWORD(-1);
     Err := WNetEnumResource(Enum, Entries, @ScanRes, Size);
     if Err = NO_ERROR then
     try
       for I := 0 to Entries - 1 do
       begin
         if StopScan then Exit; //
         with ScanRes[i] do     //
         begin
           if dwDisplayType = RESOURCEDISPLAYTYPE_SERVER then
             if lpRemoteName = CompName then // если нашли наш компьютер...
             begin
               Result :=  lpComment;
               StopScan := True;        //
               Exit;
             end;
           if dwDisplayType <> RESOURCEDISPLAYTYPE_SERVER then  //
             Scan(ScanRes[i], False);
         end;
       end;
     finally
       WNetCloseEnum(Enum);
     end
     else
       if Err <> ERROR_NO_MORE_ITEMS then  // Нет элементов для отображения...
         MessageDlg(SysErrorMessage(GetLastError), mtError, [mbOK], 0);
    end
    else
   end;
begin
  //.
  Result := RES_UNKNOWN;
```

```
  if CompName = RES_UNKNOWN then Exit;    //
                                  CompName := '\\' + CompName;           //

  StopScan := False;    //
  Scan(TmpRes, True);
  // Результат...
  if Result = '' then Result := RES_COM_NO;
end;
function TForm1.GetProvider(const CompName: String): String;
var
  Buffer: array [0..255] of Char;
  Size: DWORD;
begin
  Size := SizeOf(Buffer);
  if WNetGetProviderName(WNNC_NET_LANMAN, @Buffer, Size) <> NO_ERROR then
    Result := RES_COM_NO
  else
    Result := String(Buffer);
end;
function TForm1.GetMacFromIP(const IP: String): String;
  function GetMAC(Value: TMacAddress; Length: DWORD): String;
  var
    I: Integer;
  begin
    if Length = 0 then Result := '00-00-00-00-00-00' else
    begin
      Result := '';
      for i:= 0 to Length -2 do
        Result := Result + IntToHex(Value[i], 2) + '-';
      Result := Result + IntToHex(Value[Length-1], 2);
    end;
  end;
  function GetDottedIPFromInAddr(const InAddr: Integer): String;
  begin     Result := '';
    Result := IntToStr(FOURTH_IPADDRESS(InAddr));
    Result := Result + '.' + IntToStr(THIRD_IPADDRESS(InAddr));
    Result := Result + '.' + IntToStr(SECOND_IPADDRESS(InAddr));
    Result := Result + '.' + IntToStr(FIRST_IPADDRESS(InAddr));
  end;

var
  Table: TMibIPNetTable;
  Size: Integer;
  CatchIP: String;
  Err, I: Integer;
Begin   Result := RES_UNKNOWN;
  Size := SizeOf(Table);
  Err := GetIpNetTable(@Table, @Size, False);
  if Err <> NO_ERROR then
  begin
    Exit;
  end;
  for I := 0 to Table.dwNumEntries - 1 do     //
```

```
  begin
    CatchIP := GetDottedIPFromInAddr(Table.Table[I].dwAddr);
    if CatchIP = IP then
// MAC ...
    begin
      Result := GetMAC(Table.Table[I].bPhysAddr, Table.Table[I].dwPhysAddrLen);
      Break;     end;   end; end;
function TForm1.GetShares(const CompName: String): String;
type TShareInfo1Array = array of TShareInfo1;
var
  entriesread, totalentries: DWORD;
  Info: Pointer;
  I: Integer;
  CN: PWideChar;
begin
  CN := StringToOleStr(CompName);
  if NetShareEnum(CN, 1, Info, DWORD(-1), @entriesread,
    @totalentries, nil) = 0 then
    try       if entriesread > 0 then
        for I := 0 to entriesread - 1 do
          Result := Result + TShareInfo1Array(@(Info^))[I].shi1_netname + ' ';
    finally
      NetApiBufferFree(Info);
    end; end;

function TForm1.GetDomainServer(const DomainName: String): String;
var
  Domain: PWideChar;
  Server: PWideChar;
begin
  GetMem(Domain, MAX_PATH);
  try
    StringToWideChar(DomainName, Domain, MAX_PATH);
    if NetGetAnyDCName(nil, Domain, @Server)= NO_ERROR then
    try       Result := WideCharToString(Server);
              finally
      NetApiBufferFree(Server);
           end;
  finally
    FreeMem(Domain, MAX_PATH);
  end; end;
function TForm1.GetGroups(DomainServer: String; UserName: String): String;
type
  TGroupUsersInfoArray = array of TGroupUsersInfo0;
var
  Info: PGroupUsersInfo0;
  Sn, Un: PWideChar;
  entriesread, totalentries: DWORD;
  I, A, B, Size: Integer;
  P: Pointer;
begin
  Sn := StringToOLEStr(DomainServer);
```

```
  Un := StringToOleStr(UserName);
  if NetUserGetGroups(Sn, Un, 0, @Info, DWORD(-1), entriesread, totalentries) = NO_ERROR
then
  try
    if entriesread > 0 then
      for I := 0 to entriesread - 1 do
        Result := Result + TGroupUsersInfoArray(@(Info^))[I].grui0_name + ' ';
  finally
    NetApiBufferFree(Info);
  end;
end;
function TForm1.PortStateToStr(const State: DWORD): String;
begin
  case State of
    MIB_TCP_STATE_CLOSED: Result := 'CLOSED';
    MIB_TCP_STATE_LISTEN: Result := 'LISTEN';
    MIB_TCP_STATE_SYN_SENT: Result := 'SYN SENT';
    MIB_TCP_STATE_SYN_RCVD: Result := 'SYN RECEIVED';
    MIB_TCP_STATE_ESTAB: Result := 'ESTABLISHED';
    MIB_TCP_STATE_FIN_WAIT1: Result := 'FIN WAIT 1';
    MIB_TCP_STATE_FIN_WAIT2: Result := 'FIN WAIT 2';
    MIB_TCP_STATE_CLOSE_WAIT: Result := 'CLOSE WAIT';
    MIB_TCP_STATE_CLOSING: Result := 'CLOSING';
    MIB_TCP_STATE_LAST_ACK: Result := 'LAST ACK';
    MIB_TCP_STATE_TIME_WAIT: Result := 'TIME WAIT';
    MIB_TCP_STATE_DELETE_TCB: Result := 'DELETE TCB';
  else
    Result := 'UNKNOWN';
  end; end;
procedure TForm1.GetMemoryInfo;
var
 MemInfo : TMemoryStatus;
begin
 MemInfo.dwLength := Sizeof (MemInfo);
 GlobalMemoryStatus (MemInfo);
 TotalPhys.caption:=inttostr(MemInfo.dwTotalPhys div 1048576) + ' Mb';
 AvailPhys.caption:=inttostr(MemInfo.dwAvailPhys div 1048576) + ' Mb';
 TotalPage.caption:=inttostr(MemInfo.dwTotalPageFile div 1048576) + ' Mb';
 AvailPage.caption:=inttostr(MemInfo.dwAvailPageFile div 1048576) + ' Mb';
 AdvProgressBar1.Position := MemInfo.dwAvailPhys div (MemInfo.dwTotalPhys div 100);
 AdvProgressBar2.Position := MemInfo.dwAvailPageFile div (MemInfo.dwTotalPageFile div 100);
end;
procedure TForm1.GetCompInfo;
var
 SystemIniFile:TIniFile;
 RegFile:TRegIniFile;
 PathArray : array [0..255] of char;
 OSVersion: TOSVersionInfo;
begin
 //Computer
 SystemIniFile:=TIniFile.Create('\\'+IPst+'\System.ini');
 ComputerLabel.Caption:=SystemIniFile.ReadString('boot.description', 'system.drv', 'Unknown');
 SystemIniFile.Free;
```

```
 RegFile:=TRegIniFile.Create('Software');
 RegFile.RootKey:=HKEY_LOCAL_MACHINE;
 RegFile.OpenKey('hardware',false);
 RegFile.OpenKey('DESCRIPTION',false);
 RegFile.OpenKey('System',false);
 RegFile.OpenKey('CentralProcessor',false);
 ProcessorLabel.Caption:=RegFile.ReadString('0','Identifier','Unknown');
 MMXIdentifierLabel.Caption:=RegFile.ReadString('0','MMXIdentifier','Unknown');
 VendorIdentifierLabel.Caption:=RegFile.ReadString('0','VendorIdentifier','Unknown');
//OS
 OSVersion.dwOSVersionInfoSize := SizeOf(OSVersion);
 if GetVersionEx(OSVersion) then
  begin
   VersionLabel.Caption:=          Format('%d.%d          (%d.%s)',[OSVersion.dwMajorVersion,
OSVersion.dwMinorVersion,(OSVersion.dwBuildNumber              and              $FFFF),
OSVersion.szCSDVersion]);
   case OSVersion.dwPlatformID of
    VER_PLATFORM_WIN32s:       VersionNumberLabel.Caption := 'Windows 3.1';
    VER_PLATFORM_WIN32_WINDOWS: VersionNumberLabel.Caption := 'Windows 95';
    VER_PLATFORM_WIN32_NT:     VersionNumberLabel.Caption := 'Windows NT';
   else                  VersionNumberLabel.Caption := '';
   end;   end;
 RegFile.CloseKey;
 RegFile.OpenKey('SOFTWARE',false);
 RegFile.OpenKey('Microsoft',false);
 RegFile.OpenKey('Windows',false);
 OSNameLabel.Caption:=RegFile.ReadString('CurrentVersion','ProductName','Unknown');
RegisteredOrganizationLabel.Caption:=RegFile.ReadString('CurrentVersion','RegisteredOrganizati
on','Unknown');
RegisteredOwnerLabel.Caption:=RegFile.ReadString('CurrentVersion','RegisteredOwner','Unknow
n');
 SerNumberEdit.Caption:=RegFile.ReadString('CurrentVersion','ProductId','Unknown');
 RegFile.Free;
 FillChar(PathArray, SizeOf(PathArray), #0);
 GetWindowsDirectory(PathArray,255);
 WindowsDirLabel.Caption:= Format('%s',[PathArray]);
 FillChar(PathArray, SizeOf(PathArray), #0);
 ExpandEnvironmentStrings('%TEMP%', PathArray, 255);
 TempDir.Caption:=Format('%s',[PathArray]);
end;
procedure TForm1.GetIPInfo;
var
 FixedInfoSize, Err, AdapterInfoSize:DWORD;
 pFixedInfo:PFIXED_INFO;
 pAdapterInfo, pAdapt:PIP_ADAPTER_INFO;
 pAddrStr:PIP_ADDR_STRING;
begin
 FixedInfoSize:=0;
 Err:=GetNetworkParams(nil, FixedInfoSize);
 if (Err<>0) and (Err<>ERROR_BUFFER_OVERFLOW) then
  begin
   HostNameLabel.Caption:='Error';
   exit;   end;
```

```
pFixedInfo:=PFIXED_INFO(GlobalAlloc(GPTR, FixedInfoSize));
GetNetworkParams(pFixedInfo, FixedInfoSize);
HostNameLabel.Caption:=StrPas(pFixedInfo.HostName);
DNSListBox.Items.Clear;
DNSListBox.Items.Add(StrPas(pFixedInfo.DnsServerList.IpAddress.S));
pAddrStr:=pFixedInfo.DnsServerList.Next;
while (pAddrStr<>nil) do
 begin
   DNSListBox.Items.Add(StrPas(pAddrStr.IpAddress.S));
  pAddrStr:=pAddrStr.Next;
 end;
case pFixedInfo.NodeType of
 1: NodeTypeLabel.Caption:='Broadcast';
 2: NodeTypeLabel.Caption:='Peer to peer';
 4: NodeTypeLabel.Caption:='Mixed';
 8: NodeTypeLabel.Caption:='Hybrid';
end;
NetBIOSScopeLabel.Caption:=pFixedInfo.ScopeId;
if pFixedInfo.EnableRouting>0 then
 IPRoutingLabel.Caption:='Yes'
else
 IPRoutingLabel.Caption:='No';
if pFixedInfo.EnableProxy>0 then
 WINSProxyLabel.Caption:='Yes'     else
 WINSProxyLabel.Caption:='No';
if pFixedInfo.EnableDns>0 then
 NetBIOSResolutionLabel.Caption:='Yes'
else
 NetBIOSResolutionLabel.Caption:='No';
//Get Adapter Info
AdapterCB.Items.Clear;
AdapterInfoSize:=0;
Err:=GetAdaptersInfo(nil, AdapterInfoSize);
if (Err<>0) and (Err<>ERROR_BUFFER_OVERFLOW) then
 begin
  AdapterCB.Items.Add('Error');
  exit;
 end;
pAdapterInfo := PIP_ADAPTER_INFO(GlobalAlloc(GPTR, AdapterInfoSize));
GetAdaptersInfo(pAdapterInfo, AdapterInfoSize);
pAdapt := pAdapterInfo;
while pAdapt<>nil do
 begin
  case pAdapt.Type_ of
   MIB_IF_TYPE_ETHERNET:
    AdapterCB.Items.Add('Ethernet adapter '+pAdapt.AdapterName);
   MIB_IF_TYPE_TOKENRING:
    AdapterCB.Items.Add('Token Ring adapter '+pAdapt.AdapterName);
   MIB_IF_TYPE_FDDI:
    AdapterCB.Items.Add('FDDI adapter '+pAdapt.AdapterName);
   MIB_IF_TYPE_PPP:
    AdapterCB.Items.Add('PPP adapter '+pAdapt.AdapterName);
   MIB_IF_TYPE_LOOPBACK:
```

```
     AdapterCB.Items.Add('Loopback adapter '+pAdapt.AdapterName);
    MIB_IF_TYPE_SLIP:
     AdapterCB.Items.Add('Slip adapter '+pAdapt.AdapterName);
    MIB_IF_TYPE_OTHER:
     AdapterCB.Items.Add('Other adapter '+pAdapt.AdapterName);
   end;
   pAdapt := pAdapt.Next;
  end;
 GlobalFree(Cardinal(pFixedInfo));
end;
procedure TForm1.Button2Click(Sender: TObject);
begin
 PageControl3.ActivePageIndex:=0;
 GetMemoryInfo;
 GetCompInfo;
 GetIPInfo;
 GetAdapterInfo;
 UpdateDisk;
end;
procedure TForm1.GetAdapterInfo;
var
 Err, AdapterInfoSize:DWORD;
 pAdapterInfo, pAdapt:PIP_ADAPTER_INFO;
 Str:String;
 i:Integer;
 pAddrStr:PIP_ADDR_STRING;
begin
 AdapterTypeLabel.Caption:='';
 AdapterNameLabel.Caption:='';
 DescriptionLabel.Caption:='';
 PhysicaladdressLabel.Caption:='';
 IPListView.Clear;
 GatewayLabel.Caption:='';
 DHCPLabel.Caption:='';
 DHCPServerLabel.Caption:='';
 SecondaryWINSLabel.Caption:='';
 PrimaryWINSLabel.Caption:='';
 AdapterInfoSize:=0;
 Err:=GetAdaptersInfo(nil, AdapterInfoSize);
 if (Err<>0) and (Err<>ERROR_BUFFER_OVERFLOW) then
  begin
   AdapterCB.Items.Add('Error');
   exit;   end;
 pAdapterInfo := PIP_ADAPTER_INFO(GlobalAlloc(GPTR, AdapterInfoSize));
 GetAdaptersInfo(pAdapterInfo, AdapterInfoSize);
 pAdapt := pAdapterInfo;
 while pAdapt<>nil do
  begin
   case pAdapt.Type_ of
    MIB_IF_TYPE_ETHERNET:
     Str:='Ethernet adapter ';
    MIB_IF_TYPE_TOKENRING:
     Str:='Token Ring adapter ';
```

```
    MIB_IF_TYPE_FDDI:
     Str:='FDDI adapter ';
    MIB_IF_TYPE_PPP:
     Str:='PPP adapter ';
    MIB_IF_TYPE_LOOPBACK:
     Str:='Loopback adapter ';
    MIB_IF_TYPE_SLIP:
     Str:='Slip adapter ';
    MIB_IF_TYPE_OTHER:
     Str:='Other adapter ';
   end;
   if Str+pAdapt.AdapterName<>AdapterCB.Text then
    begin
     pAdapt := pAdapt.Next;
     Continue;
    end;
    AdapterTypeLabel.Caption:=Str;
    AdapterNameLabel.Caption:=AdapterCB.Text;
    DescriptionLabel.Caption:=pAdapt.Description;
    Str:='';
    for i:=0 to pAdapt.AddressLength-1 do
     begin
      Str:=Str+IntToHex(pAdapt.Address[i],2);
      if i<>Integer(pAdapt.AddressLength-1) then
       Str:=Str+'-';
     end;
    PhysicaladdressLabel.Caption:=Str;
    pAddrStr:=@pAdapt.IpAddressList;
    while pAddrStr<>nil do
     begin
      with IPListView.Items.Add do
       begin
        Caption:=pAddrStr.IpAddress.S;
        SubItems.Add(pAddrStr.IpMask.S);
       end;
      pAddrStr := pAddrStr.Next;
     end;
    if pAdapt.DhcpEnabled=0 then
     DHCPLabel.Caption:='no'
    else
     DHCPLabel.Caption:='yes';
    DHCPServerLabel.Caption:=pAdapt.DhcpServer.IpAddress.S;
    PrimaryWINSLabel.Caption:=pAdapt.PrimaryWinsServer.IpAddress.S;
    SecondaryWINSLabel.Caption:=pAdapt.SecondaryWinsServer.IpAddress.S;
    GatewayLabel.Caption:=pAdapt.GatewayList.IpAddress.S;
    break;
  end;
 GlobalFree(Cardinal(pAdapterInfo));
end;
procedure TForm1.UpdateDisk;
var
 lpRootPathName          : PChar;
 lpVolumeNameBuffer      : PChar;
```

```
 nVolumeNameSize          : DWORD;
 lpVolumeSerialNumber     : DWORD;
 lpMaximumComponentLength : DWORD;
 lpFileSystemFlags        : DWORD;
 lpFileSystemNameBuffer   : PChar;
 nFileSystemNameSize      : DWORD;
 FSectorsPerCluster: DWORD;
 FBytesPerSector   : DWORD;
 FFreeClusters     : DWORD;
 FTotalClusters    : DWORD;
begin
 lpVolumeNameBuffer      := '';
 lpVolumeSerialNumber    := 0;
 lpMaximumComponentLength:= 0;
 lpFileSystemFlags       := 0;
 lpFileSystemNameBuffer  := '';
 try
  GetMem(lpVolumeNameBuffer, MAX_PATH + 1);
  GetMem(lpFileSystemNameBuffer, MAX_PATH + 1);
  nVolumeNameSize := MAX_PATH + 1;
  nFileSystemNameSize := MAX_PATH + 1;
  lpRootPathName := PChar(DriveComboBox1.Drive+':\');
  if GetVolumeInformation( lpRootPathName, lpVolumeNameBuffer,
      nVolumeNameSize, @lpVolumeSerialNumber, lpMaximumComponentLength,
      lpFileSystemFlags, lpFileSystemNameBuffer, nFileSystemNameSize )
   then
     begin
      VolumeName.Caption    := lpVolumeNameBuffer;
      VolumeSerial.Caption        :=    IntToHex(HIWord(lpVolumeSerialNumber),    4)    +    '-'    +
IntToHex(LOWord(lpVolumeSerialNumber), 4);
      FileSystemName.Caption:= lpFileSystemNameBuffer;
      GetDiskFreeSpace( PChar(DriveComboBox1.Drive+':\'), FSectorsPerCluster, FBytesPerSector,
FFreeClusters, FTotalClusters);
     end;
 finally
  FreeMem(lpVolumeNameBuffer);
  FreeMem(lpFileSystemNameBuffer);
  end;
 SectorsPerCluster.Caption:=IntToStr(FSectorsPerCluster);
 BytesPerSector.Caption:=IntToStr(FBytesPerSector);
end;
procedure TForm1.DriveComboBox1Change(Sender: TObject);
begin
  UpdateDisk;
end;
procedure TForm1.AdapterCBChange(Sender: TObject);
begin
  GetAdapterInfo;
end;
procedure TForm1.FormShow(Sender: TObject);
begin
  PageControl3.ActivePageIndex:=0;
  GetMemoryInfo;
```

```
  GetCompInfo;
  GetIPInfo;
  GetAdapterInfo;
  UpdateDisk;
end;
procedure TForm1.Button31Click(Sender: TObject);
var
  Size: DWORD;
  TCPTable: PTMibTCPTable;
  UDPTable: PTMibUdpTable;
  I: DWORD;
begin
  Memo1.Clear;
  GetMem(TCPTable, SizeOf(TMibTCPTable));
  try
    Size := 0;

    if GetTcpTable(TCPTable, Size, True) <> ERROR_INSUFFICIENT_BUFFER then Exit;
  finally
    FreeMem(TCPTable);
  end;
  GetMem(TCPTable, Size);
  try
    if GetTcpTable(TCPTable, Size, True) = NO_ERROR then
    begin
      Memo1.Lines.Add('');
      Memo1.Lines.Add('            АНАЛІЗ ПО TCP ПРОТОКОЛУ');
      Memo1.Lines.Add('');
      Memo1.Lines.Add(Format('%15s: | %5s %-12s', ['Хост', 'Порт', 'Состояние']));
      Memo1.Lines.Add('=============================================');
    for I := 0 to TCPTable^.dwNumEntries - 1 do
      Memo1.Lines.Add(Format('%15s:                    |                %5d                %s',
[inet_ntoa(in_addr(TCPTable^.Table[I].dwLocalAddr)),
       htons(TCPTable^.Table[I].dwLocalPort), PortStateToStr(TCPTable^.Table[I].dwState)]));
    end;
  finally
    FreeMem(TCPTable);
  end;
  GetMem(UDPTable, SizeOf(TMibUDPTable));
  try
    Size := 0;
    if GetUdpTable(UDPTable, Size, True) <> ERROR_INSUFFICIENT_BUFFER then Exit;
  finally
    FreeMem(UDPTable);
  end;
  GetMem(UDPTable, Size);
  try
    if GetUdpTable(UDPTable, Size, True) = NO_ERROR then
    begin
      Memo1.Lines.Add('');
      Memo1.Lines.Add('            АНАЛІЗ ПО UDP ПРОТОКОЛУ');
      Memo1.Lines.Add('');
      Memo1.Lines.Add(Format('%15s: | %5s ', ['Хост', 'Порт']));
```

```
    Memo1.Lines.Add('=======================================');
   for I := 0 to UDPTable^.dwNumEntries - 1 do
    Memo1.Lines.Add(Format('%15s:                    |                    %5d',
[inet_ntoa(in_addr(UDPTable^.Table[I].dwLocalAddr)),
     htons(UDPTable^.Table[I].dwLocalPort)]));
   end;
  finally
   FreeMem(UDPTable);
   Memo1.Lines.Delete(0);
   Memo1.Lines.Insert(0,'');
  end; end;
procedure TForm1.Button21Click(Sender: TObject);
  function ProcessPIDToName(const hProcessSnap: THandle; ProcessId: DWORD): String;
  var
   processEntry: TProcessEntry32;
  begin
   Result := '';
   FillChar(processEntry, SizeOf(TProcessEntry32), #0);
   processEntry.dwSize := SizeOf(TProcessEntry32);
   if not Process32First(hProcessSnap, processEntry) then Exit;
   repeat
    if processEntry.th32ProcessID = ProcessId then
    begin
     Result := String(processEntry.szExeFile);
     Exit;     end;
   until not Process32Next(hProcessSnap, processEntry);
  end;
var
  TCPExTable: PTMibTCPExTable;
  UDPExTable: PTMibUdpExTable;
  I: DWORD;
  hProcessSnap: THandle;
begin
  Memo1.Clear;

  hProcessSnap := CreateToolhelp32Snapshot(TH32CS_SNAPPROCESS, 0);
  if (hProcessSnap = INVALID_HANDLE_VALUE) then
  begin
   Memo1.Lines.Add('');
   Memo1.Lines.Add('CreateToolhelp32Snapshot failed');
   Exit;
  end;
  try
   if AllocateAndGetTcpExTableFromStack(@TCPExTable, False, GetProcessHeap, 2, 2) =
NO_ERROR then
   try
    Memo1.Lines.Add('');
    Memo1.Lines.Add('          АНАЛІЗ ПО TCP ПРОТОКОЛУ');
    Memo1.Lines.Add('');
    Memo1.Lines.Add(Format('%15s: | %5s | %-12s | %20s | (%s)', ['Хост', 'Порт', 'Состояние',
'Имя процесса', 'ИД']));
    Memo1.Lines.Add('=================================================');
    for I := 0 to TCPExTable^.dwNumEntries - 1 do
```

```
      Memo1.Lines.Add(Format('%15s: | %5d | %-12s | %20s | (%d)',
        [inet_ntoa(in_addr(TCPExTable^.Table[I].dwLocalAddr)),
        htons(TCPExTable^.Table[I].dwLocalPort),
        PortStateToStr(TCPExTable^.Table[I].dwState),
        ProcessPIDToName(hProcessSnap, TCPExTable^.Table[I].dwProcessID),
        TCPExTable^.Table[I].dwProcessID]));
   finally
    GlobalFreePtr(TCPExTable);
   end;
   if AllocateAndGetUdpExTableFromStack(@UDPExTable, False, GetProcessHeap, 2, 2) =
NO_ERROR then
   try
    Memo1.Lines.Add('');
    Memo1.Lines.Add('            АНАЛІЗ ПО UDP ПРОТОКОЛУ');
    Memo1.Lines.Add('');
    Memo1.Lines.Add(Format('%15s: | %5s |   %19s | (%s)', ['Хост', 'Порт',  'Имя процесса',
'ИД']));
    Memo1.Lines.Add('================================================');

    for I := 0 to UDPExTable^.dwNumEntries - 1 do
     Memo1.Lines.Add(Format('%15s: | %5d | %20s | (%d)',
      [inet_ntoa(in_addr(UDPExTable^.Table[I].dwLocalAddr)),
      htons(UDPExTable^.Table[I].dwLocalPort),
      ProcessPIDToName(hProcessSnap, UDPExTable^.Table[I].dwProcessID),
      UDPExTable^.Table[I].dwProcessID]));
   finally
    GlobalFreePtr(UDPExTable);
   end;
  finally
   CloseHandle(hProcessSnap);
   Memo1.Lines.Delete(0);
   Memo1.Lines.Insert(0,'');
  end; end;
procedure TForm1.memInfoClick(Sender: TObject);
var   Line: Integer ;
begin
  with (Sender as TMemo) do begin
    Line      := Perform(EM_LINEFROMCHAR, SelStart, 0);
    SelStart  := Perform(EM_LINEINDEX, Line, 0);
    SelLength := Length(Lines[Line]);
  end; end;
procedure TForm1.Memo1Click(Sender: TObject);
var   Line: Integer ;
begin
  with (Sender as TMemo) do begin
    Line      := Perform(EM_LINEFROMCHAR, SelStart, 0);
    SelStart  := Perform(EM_LINEINDEX, Line, 0);
    SelLength := Length(Lines[Line]);
  end; end;
procedure TForm1.AdvStringGrid2DrawCell(Sender: TObject; ACol,
  ARow: Integer; Rect: TRect; State: TGridDrawState);
var s: string;
begin  with Sender as TAdvStringGrid do begin
```

```
    s:=cells[acol,arow]; //сохраняем текст из ячейки
    canvas.FillRect(rect);
    DrawText(canvas.handle,pchar(s),-1,Rect, DT_CENTER or DT_WORDBREAK)
end;
end;
procedure TForm1.RadioButton1Click(Sender: TObject);
var
  i,j,i_max,i_min:Integer;
  ar: array [1..7,1..4] of real;
  str : string;
  res1_max,res2_max,res1_min,res2_min:real;
begin   for i := 1 to 7 do
    for j:=1 to 4 do begin
      str:=replace(form1.AdvStringGrid4.Cells[j-1,i],'.',',');
      ar[i,j]:=StrToFloat(str);
    end;
  res1_max:=0; res1_min:=1000;
  i_min:=0; i_max:=0;
  for i := 1 to 7 do begin
    res2_max:=0; res2_min:=0;
    for j:=1 to 4 do begin
       res2_max:=res2_max+ar[i,j];
       res2_min:=res2_min+ar[i,j];
    end;
    if res2_max>=res1_max then begin
      res1_max:=res2_max; i_max:=i;      end;
    if res2_min<=res1_min then begin
      res1_min:=res2_min; i_min:=i;
    end;   end;
end;
procedure TForm1.RadioButton2Click(Sender: TObject);
var
  i,j,i_max,i_min:Integer;
  ar: array [1..7,1..4] of real;
  ar1: array [1..7] of real;
  str : string;
  res1_max,res2_max,res1_min,res2_min:real;
begin
  for i := 1 to 7 do
    for j:=1 to 4 do begin
      str:=replace(form1.AdvStringGrid4.Cells[j-1,i],'.',',');
      ar[i,j]:=StrToFloat(str);
    end;
  for i := 1 to 7 do begin
    ar1[i]:=1000;
    for j:=1 to 4 do if ar1[i]>ar[i,j]then ar1[i]:=ar[i,j]
  end;
  i_max:=1; i_min:=1;
  i_min:=0; i_max:=0;
  res1_max:=0; res1_min:=1000;
  for i := 1 to 7 do begin
    if ar1[i]>=res1_max then begin
      res1_max:=ar1[i];
```

```
      i_max:=i;
    end;
    if ar1[i]<=res1_min then begin
      res1_min:=ar1[i];
      i_min:=i;
    end;   end;
  if i_max<>i_min then
матриці';
end;
procedure TForm1.RadioButton3Click(Sender: TObject);
var
  i,j,i_max,i_min:Integer;
  ar: array [1..7,1..4] of real;
  ar1: array [1..7] of real;
  str : string;
  res1_max,res2_max,res1_min,res2_min:real;
begin   for i := 1 to 7 do
    for j:=1 to 4 do begin
      str:=replace(form1.AdvStringGrid4.Cells[j-1,i],'.',',');
      ar[i,j]:=StrToFloat(str);
    end;
  for i := 1 to 7 do begin  ar1[i]:=0;
    for j:=1 to 4 do if ar1[i]<ar[i,j]then ar1[i]:=ar[i,j]
  end;
  for i := 1 to 7 do  for j:=1 to 4 do ar[i,j]:=ar1[i]-ar[i,j];
  for i := 1 to 7 do begin      ar1[i]:=0;
    for j:=1 to 4 do if ar1[i]<ar[i,j]then ar1[i]:=ar[i,j]
  end;
  i_min:=0; i_max:=0;
  res1_max:=0; res1_min:=1000;
  for i := 1 to 7 do begin
    if ar1[i]>res1_max then begin
      res1_max:=ar1[i];
      i_max:=i;
    end;
    if ar1[i]<res1_min then begin  res1_min:=ar1[i];
      i_min:=i;
    end;
  end;
end;
procedure TForm1.RadioButton4Click(Sender: TObject);
var   i,j,i_max,i_min:Integer;
  ar: array [1..7,1..4] of real;
  ar1,ar2,ar3: array [1..7] of real;
  str : string;
  res1_max,res2_max,res1_min,res2_min,a:real;
begin   for i := 1 to 7 do  for j:=1 to 4 do begin
str:=replace(form1.AdvStringGrid4.Cells[j-1,i],'.',',');
      ar[i,j]:=StrToFloat(str);
    end;
    //max
  for i := 1 to 7 do begin      ar1[i]:=0;
    for j:=1 to 4 do if ar1[i]<ar[i,j]then ar1[i]:=ar[i,j]
```

```
  end;
     //min
  for i := 1 to 7 do begin    ar2[i]:=1000;
    for j:=1 to 4 do if ar2[i]>ar[i,j]then ar2[i]:=ar[i,j]
  end;
  for i := 1 to 7 do ar3[i]:=a*ar2[i]+(1-a)*ar1[i];
  i_min:=0; i_max:=0;
  res1_max:=0; res1_min:=1000;
  for i := 1 to 7 do begin  if ar3[i]>res1_max then begin res1_max:=ar3[i];
       i_max:=i;      end;
     if ar3[i]<res1_min then begin
       res1_min:=ar3[i];
       i_min:=i;
     end;    end;
end;
procedure TForm1.AdvStringGrid2SelectCell(Sender: TObject; ACol,
  ARow: Integer; var CanSelect: Boolean);
Begin StatusBar1.Panels[0].Text := '';
  StatusBar1.Panels[1].Text :=arOp[ARow,ACol];
end;
procedure TForm1.AdvStringGrid4SelectCell(Sender: TObject; ACol,
  ARow: Integer; var CanSelect: Boolean);
Begin StatusBar1.Panels[0].Text := '';
  StatusBar1.Panels[1].Text :=arOp[ARow,ACol+1];
end;
procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin   WriteFile();
end;
procedure TForm1.AdvGlowButton1Click(Sender: TObject);
var
  i,j:byte; str:string; res:real;
begin   for i := 1 to 7 do
     for j:=1 to 4 do begin   str:=replace(AdvStringGrid2.Cells[j,i],'.',',');
       res:=StrToFloat(str)*
            StrToFloat(AdvStringGrid1.Cells[1,j-1])*
            StrToFloat(AdvStringGrid3.Cells[1,i-1]);
       AdvStringGrid4.Cells[j-1,i]:=FloatToStr(round(res*1000)/1000);
     end; end;
procedure TForm1.AdvGlowButton2Click(Sender: TObject);
var
  Col, Row: Integer; X, Y : Double; Sg : TAdvStringGrid; S : String; i, j: Integer;
begin   for i:=1 to N do  O[i]:= StrToFloat(AdvStringGrid7.Cells[1,i]);
  for i:=1 to N do
    for j:=1 to N do
      F[i,j]:= StrToFloat(AdvStringGrid8.Cells[i,j]);
  for i:=1 to N do  for j:=1 to N do   OO[i,j]:=O[i];
  for i:=1 to N do begin Res_S[i] := RoundFloat(M(i).S,2);
    Res_Q[i] := RoundFloat(M(i).Q,2);
  end;
  for i:=1 to N do begin
    AdvStringGrid9.Cells[1,i] := Floattostr(Res_S[i]);
    AdvStringGrid9.Cells[2,i] := Floattostr(Res_Q[i]);
  end;
```

```
 Sg := AdvStringGrid9;
 Series1.Clear;
 for Row := Sg.FixedCols to Sg.RowCount-1 do begin
   X := StrToFloatDef( Sg.Cells[1, Row ], 0 );
   Y := StrToFloatDef( Sg.Cells[2, Row ], 0 );
   S := '(' + FloatToStr(X) + ', ' + FloatToStr(Y) + ')';
   Series1.AddXY(X, Y, S);
   end;
end;
end.
```

НАВЧАЛЬНЕ ВИДАННЯ

Лахно Валерій Анатолійович,

Гусєв Борис Семенович,

Касаткін Дмитро Юрійович,

Хорольська Карина Вікторівна

ЗАХИСТ ІНФОРМАЦІЇ В КОМП'ЮТЕРНИХ СИСТЕМАХ І КІБЕРБЕЗПЕКА

ЧАСТИНА 1

Навчальний посібник

Підписано до друку 02.10.23 Формат 60х84\16
Ум. друк. арк. 17,4 Наклад 100 прим. Зам. № 230518

Видавець і виготовлювач Національний університет біоресурсів
і природокористування України,
вул. Героїв Оборони, 15, м. Київ, 03041.
Свідоцтво суб'єкта видавничої справи
ДК № 4097 від 17.06.2011