МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

ВИПУСКНА РОБОТА

на тему:

«Експертна система оцінки захищеності інформації в комп'ютерних системах від несанкціонованого доступу»

Завідувач випускаючої кафедри — **Довбиш А.С.**

Керівник роботи — **Шелехов І.В.**

Студента групи КБ – 61 — **Гончаренка І.Я.**

СУМИ 2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

Кафедра комп'ютерних наук

Затверджую_____
Зав. кафедрою Довбиш А.С.
"_____" _____2020 р.

ЗАВДАННЯ

до випускної роботи

Студента четвертого курсу, групи КБ-61 спеціальності "Кібербезпека" денної форми навчання Гончаренка Іллі Ярославовича.

Тема: "Експертна система оцінки захищеності інформації в комп'ютерних системах від несанкціонованого доступу "

Затверджена наказом по СумДУ
№_____ от _____2020 р.

Зміст пояснювальної записки: 1) аналітичний огляд систем захисту інформації; 2) постановка завдання й формування завдань дослідження; 3) вибір методу розв'язання задачі; 5) інформаційне та програмне забезпечення експертної системи; 6) аналіз результатів.

Дата видачі завдання "_____"_____2020 р.
Керівник випускної роботи _____ Шелехов І.В.
Завдання прийняв до виконання _____ Гончаренко І.Я.

РЕФЕРАТ

Записка: 49 стор., 9 рис., 1 додаток, 11 джерел.

Об’єкт дослідження — процес оцінки захищеності комп’ютерних систем.

Мета роботи — розробка експертної системи для оцінки захищеності комп’ютерних систем.

Результати — Створена експертна система для оцінювання захищеності комп’ютерних систем від несанкціонованого доступу. Система є цілком працездатною, про що свідчить успішне тестування.

ЕКСПЕРТНА СИСТЕМА, НЕСАНКЦІОНОВАНИЙ ДОСТУП,
ОЦІНКА ЗАХИЩЕНОСТІ, CLIPS

ЗМІСТ

ВСТУП

Використання штучного інтелекту в системі інформаційної та/або кібербезпеки безпеки може бути використане для зменшення постійно зростаючих загроз, з якими стикається глобальний бізнес. У багатьох галузях промисловості додатки, що використовують машинне навчання, а також штучний інтелект (ШІ), все ширше використовуються, оскільки збір даних, можливості зберігання даних та обчислювальна потужність зростають. У реальному часі людині важко оперувати величезною кількістю даних. За допомогою машинного навчання, а також штучного інтелекту, обробка таких даних може бути виконана за мілісекунди, в результаті чого підприємство може легко ідентифікувати загрозу, оцінити рівень власної захищеності від такої загрози, а також і відновитися після загрози. Очевидно, що протидія сучасному шкідливому програмну забезпеченню та кіберзброї може бути ефективною лише при застосуванні сучасних інтелектуальних технологій при розробці та впровадженні систем інформаційного та/або кіберзахисту .

1 АНАЛІТИЧНИЙ ОГЛЯД І ПОСТАНОВКА ЗАДАЧІ

1.1 Критерії оцінки захищеності інформації в комп’ютерних системах від несанкціонованого доступу

1.1.1 Основні положення

Критерії оцінки захищеності - це набір методологічних правил, що визначають певні вимоги до захисту інформації в комп'ютерних системах(КС) від несанкціонованого доступу. Опираючись на багатозначність даного поняття можна стверджувати, що одним з видів використання критеріїв є оцінка захищеності інформації в КС або рішення що до придатності інформації для обробки, яка підлягає захисту.

Також ці критерії слугують для реалізації КС, якій мають достатній рівень захисту відповідно до потреб і створення засобів захисту від несанкціонованого доступу.

При використанні критеріїв з’являються нові елементи для захисту КС, а саме:

- Шкала для порівняння оцінок безпечності механізмів та елементів захисту інформації від несанкціонованого доступу, які інтегруються в системах.
- Певні орієнтири в яких реалізовані методології для захисту інформації, які безпосередньо використовуються при розробці КС.

Даний ресурс може використовуватись в будь якому типі систем. Це може бути сервери баз даних, сервери веб-додатків, однорідні системи, різного типу мережу, інтегровані системи та ін.

1.1.2 Побудова і структура критеріїв захищеності інформації

В процесі оцінки можливості КС забезпечувати захист інформації, яка обробляється від несанкціонованого доступу розглядаються два види вимог:

- вимоги до послуг безпеки;
- вимоги до гарантій.

Функціональні критерії розділені на чотири групи, кожна описує вимоги до послуг, що забезпечують захист від загроз. Ці загрози відносяться до одного із чотирьох основних типів.

Загрози конфіденційності. Тип загроз, які відносяться до несанкціонованого ознайомлення інформації, становлять загрози конфіденційності. Розмежовуюсь такі види конфіденційності:

- довірча конфіденційність. Цей вид конфіденційності дозволяє клієнту КС керувати потоками інформації на захищених об'єктах до інших користувачів. Рівні даної послуги ранжируються на основі повноти(цілісності) захисту і вибірковості керування.
- адміністративна конфіденційність. Дозволяє адміністратору керувати потоками інформації від захищених об'єктів до клієнтів. Рівні даної послуги ранжируються на базі повноти(цілісності) захисту і вибірковості управління.
- повторне використання об'єктів. Дає можливість забезпечити правильність повторного використання розділених об'єктів і гарантувати, якщо розділюваний об'єкт виділяється новому клієнту або процесу, то він не має інформації, яка залишилась від попереднього клієнта або процесу.
- аналіз прихованих каналів. Метою є виявлення(знаходження) і усунення потоків інформації, які існують і не контролюються іншими послугами. Ієрархія даної послуги ранжируються на основі того, чи виконується тільки виявлення, контроль або перекриття прихованих(невидимих) каналів.
- конфіденційність при обміні (експорті/імпорті). Дає можливість забезпечити захист об'єктів від несанкціонованого доступу до інформації, що міститься під час їх експорту або імпорту через незахищене(потенційно вразливе) середовище.

Цілісність. Цей тип загроз відноситься до несанкціонованої зміни(модифікації) інформації. Види цілісності:

- довірча цілісність. Надає можливість клієнту керувати потоками інформації від інших клієнтів до захищених об'єктів, які належать його домену або середовищу.
- адміністративна цілісність. Адміністратор або спеціально авторизований клієнт має можливість керувати потоками інформації від користувачів до захищених об'єктів.
- Відкат. Надається можливість відмінити операцію або набір послідовних операцій і повернути захищений об'єкт до попереднього стану.
- цілісність при обміні. Забезпечує захист об'єктів від несанкціонованої модифікації інформації під час їх експорту чи імпорту через незахищене середовище.

Доступність. Загрози характеризуються порушенням можливості використання комп'ютерних систем або оброблюваної інформації. Види доступності:

- використання ресурсів. Надає можливість клієнтам керувати використанням послуг і ресурсів.
- стійкість до відмов. Надає певний рівень гарантій, що до доступності КС після відмови її компонента.
- гаряча заміна. Гарантує доступність КС в процесі заміни окремих модулів або компонентів.
- відновлення після збоїв. Забезпечує відновлення КС у відомий захищений стан після відмови/переривання обслуговування.

Спостереженість. Виявлення і контроль дій користувачів, управління КС становлять предмет послуг спостереженості і керованості. Види спостереженості:

- реєстрація. Створює можливість контролювати небезпечні для КС дії.

- ідентифікація та автентифікація. Визначення і перевірка особистості користувача, що намагається одержати доступ до КС.
- достовірний канал. Гарантує клієнту можливість взаємодії з комплексом засобів захисту.
- розподіл обов'язків. Зменшення потенційних збитків від навмисних або помилкових дій клієнтів або користувачів і обмеження авторитарності управління.
- цілісність комплексу засобів захисту. Визначає рівень здатності комплексу засобів захисту надати захист самої КС і гарантує спроможність управляти захищеними об'єктами.
- самотестування. На підставі перевірки дозволяє гарантувати коректність функціонування і цілісність множини функцій КС.
- автентифікація при обміні. Дає можливість одному комплексу засобів захисту ідентифікувати інший і забезпечити йому комплексну систему захисту і можливість ідентифікувати перший, перш ніж почати взаємодію.
- автентифікація відправника. Забезпечення захисту від відмови від авторства і встановлення належності об'єкта певному клієнту.
- автентифікація одержувача. Забезпечення захисту від відмови від одержання і можливість однозначно встановити факт одержання об'єкта певним клієнтом.

Окрім функціональних критеріїв, які оцінюють наявність послуг безпеки в КС також існують критерії гарантій, які дають можливість оцінити правильність реалізації послуг. Критерії гарантій складаються з вимоги до архітектури комплексу, середовища розробки, послідовності розробки, тестування комплексу засобів захисту, середовища функціонування і експлуатаційної документації.

1.2 Інтелектуальні технології захисту інформації

В межах реалізації даної задачі захисту інформації досліджується зазвичай набір формальних методів, алгоритмів або моделей, які базуються на основі програмних прототипів, що дають можливість реалізувати будь-які інтелектуальні механізми захисту:

- процес пошуку та групування інформації, яка відноситься до стану інформаційної системи, подальший аналіз з використанням модуля, який оброблює та компонує інформації з багатьох джерел.
- проактивне попередження атак;
- моніторинг та аналіз роботи системи на пошук її аномальної роботи, порушення встановленої політики безпеки користувачами і прогнозування можливих атак від правопорушників;
- використання різних типів запобіжних заходів для пасивного захисту(псевдо-важлива інформація, використання соціальної інженерії для створення пастки);
- швидке реагування на початкові етапи атаки порушника шляхом динамічного і автоматичного переналаштування механізмів захисту для відстеження та нейтралізації атаки;
- аудит та моніторинг потоку інформації в мережі з періодичною перевіркою актуальності брандмауера та політики безпеки;
- підтримка прийняття рішень з управління політиками безпеки, в тому числі щодо адаптації до наступним вторгненням і посилення критичних механізмів захисту.

1.2.1 Механізми управління захисту інформації, засновані на інтелектуальних агентах

Перспективним підходом до побудови інтелектуальних механізмів захисту інформації є технологія інтелектуальних багатоагентних систем. Цей підхід дозволяє в порівнянні з стандартними функціями покращити

ефективність захисної системи, посилити стійкість до відмов, деструкцій та пластичність самої системи.

Відповідно до даного підходу передбачається, що компоненти систем захисту інформації, спеціалізовані за типами розв'язуваних задач, тісно взаємодіють між собою для обміну інформацією та прийняттям узгоджених рішень, адаптуються до зміни трафіку, реконфігурації апаратного і програмного забезпечення, нових видів кібератак.

В рамках запропонованого підходу компоненти багатоагентної системи захисту інформації є інтелектуальні автономні програми (агенти захисту), які створюють методи захисту для забезпечення затвердженого класу захищеності. Вони дають можливість реалізувати комплексну надбудову над елементами безпеки використовуваних мережевих програмних засобів, операційних систем, підвищуючи рівень захисту системи.

В межах даного напрямку досліджень розроблені архітектури, моделі і програмні прототипи декількох многоагентних систем, в тому числі агентно-орієнтована система моделювання атак, багатоагентна система виявлення вторгнень, багатоагентна система навчання виявлення вторгнень та ін.

1.2.2 Технології дезінформації зловмисника, приховування та камуфляжу важливих ресурсів і процесів

Для захисту інформації в комп'ютерних мережах необхідно попереджати, блокувати та за певними правилами реалізовувати реагувати на зловмисників. Також важливою складовою є відволікання їх(порушників) від основних цілей, заманюючи на помилкові інформаційні об'єкти, проводити збір інформації про прийоми, тактику і мотивації зловмисників, здійснювати їх ідентифікацію та викриття.

Для виконання цих підзадач можуть бути використані так звані помилкові інформаційні системи (ПІС), звані також системами-імітаторами, обманними системами або системами-пастками.

ПІС представляють собою програмно-апаратні засоби забезпечення інформаційної безпеки, що реалізують функції приховування і камуфляжу захищаються інформаційних ресурсів, а також дезінформації порушників.

1.2.3 Підтримка життєвого циклу системи захисту інформації

У процесі використання різних механізмів захисту інформації необхідно здійснювати підтримку захищеного інформаційного середовища на різних етапах життєвого циклу, включаючи етапи його проектування, конфігурації, розгортання, функціонування та модифікації. Тому, крім створення окремих перспективних механізмів захисту, необхідно вирішувати задачу розробки моделей і методів побудови єдиної уніфікованої системи (середовища), що здійснює підтримку всього життєвого циклу, включаючи адаптивне управління політиками безпеки.

У роботі пропонується підхід до здійснення безперервного ланцюжка різних етапів життєвого циклу розподілених захищених КС.

Даний підхід передбачає реалізацію наступних механізмів:

- специфікацію політик безпеки і архітектури (або конфігурації) системи;
- трансформацію політик безпеки з метою їх уточнення (деталізації) з урахуванням опису системи;
- верифікацію політик безпеки (перевірку правильності та усунення конфліктів);
- визначення рівня безпеки і аналіз ризиків;
- моделювання поведінки системи захисту в різних умовах функціонування;
- зміна політик відповідно з необхідним рівнем безпеки і можливостями по використанню різних ресурсів і виділенню фінансових коштів на захист інформації;
- реалізацію політик безпеки в системі, в тому числі трансляції сформованих правил безпеки в параметри конфігурації і налаштування програмно-апаратного забезпечення;

- проактивний моніторинг виконання політик безпеки, в тому числі виявлення відхилень роботи користувачів від політики безпеки, виявлення вторгнень і аналіз вразливостей;
- адаптацію поведінки розподілених захищених КС і реалізованих політик безпеки відповідно до умовами функціонування.

1.3 Постановка задачі

Складність у поданні знань щодо проблеми несанкціонованого доступу до КС сприяв використанню експертних систем в галузі управління штучним інтелектом в управлінні цих мереж. Для вирішення цієї проблеми була представлена конструкція системи підтримки прийняття рішень, яка сприяє отриманню знань за допомогою опитування користувача стосовно відповідної КС, що відповідає вимогам законодавства, описані в документі НД ТЗІ 2.5-004-99. Система прийняття рішень розширює можливості звичайної експертної системи, інтегруючи моделювання функцій з знаннями експертів, заснованим на правилах, експертною оболонкою під назвою CLIPS. На відміну від попередніх експертних систем, база знань системи прийняття рішень має бути розроблена таким чином, щоб система мала можливість генерації висновку оцінювання рівня забезпечення системи безпеки від несанкціонованого доступу для будь-якого виниклого або запропонованого сценарію.

В роботі необхідно розробити експертну систему для оцінки захищеності інформації в КС від несанкціонованого доступу. Для досягнення поставленої мети необхідно виконати такі завдання:

1. Сформувати вхідний математичний опис експертної системи.
2. Сформувати базу знань, що містить вимоги щодо захищеності інформації в КС.
3. Розробити і програмно реалізувати алгоритми подання та виведення правил щодо порядку перевірки таких вимог.
4. Перевірити працездатність розробленої експертної системи.

2 ВИБІР МЕТОДУ РОЗВ'ЯЗАННЯ ЗАДАЧІ

2.1 Штучний інтелект систем інформаційної та/або кібербезпеки

Спочатку комп’ютерна безпека та штучний інтелект вважалися двома окремими утвореннями. Щоб зменшити вплив людини на процеси обробки інформації і можливі помилки при цьому, дослідники ШІ прагнули створювати інтелектуальні програми, які допомагали людині-оператору або навіть заміняли її, тоді як експерти з безпеки намагалися вирішити задачі витоку даних. З плином часу обидві галузі стали ближчими, оскільки атаки були зосереджені на тому, щоб імітувати справжнє виконання, на рівні людського клієнта, а також знижувати системний рівень. Наприклад, CAPTCHA можна вважати чудовим прикладом поєднання ШІ та безпеки. У програмі CAPTCHA клієнт повинен вводити букви чи цифри, викривлені зображення яких подано в певній послідовності на екрані. Вдосконалення методів розпізнавання зображень, які можна було використовувати як CAPTCHA, дозволило автоматизувати процес її формування. В результаті зменшувався обсяг робіт необхідний для впровадження систем захисту інформаційних комерційних ресурсів, наприклад, при бронюванні квитків через Інтернет, а галузь інформаційної та кібербезпеки почала сприяти прогресу в галузі ШІ.

Технології ШІ моделюють основні когнітивні процеси людини: здатність запам’ятовувати певну інформацію, узагальнювати її у вигляді знань і застосовувати їх в схожих умовах. Крім того ШІ надає інформаційним системам можливість змінювати та доповнювати такі знання або навчатися. Розроблені в результаті інформаційні системи стають інтелектуальними і поєднують основні переваги комп’ютерної (можливість оперувати і контролювати значні за обсягом потоки даних, здатність визначати навіть найменші зміни в них) та людської (заданість розпізнавати стани системи, в яких вона діє некоректно, і відновлювати роботу системи).

Здебільшого, ШІ можна розглядати як науку, в рамках якої розвиваються методи подання та оперування знаннями, або науку, що створює методи для розв'язання складних задач, які неможливо вирішити без моделювання когнітивних процесів притаманних людині, як, наприклад, грати у шахи.

Розглянемо основні технології ШІ, що застосовуються для вирішення складних задач в інформаційній та/або кібербезпеці.

Експертна система - це інформаційна системи для подання знань експерта в певній галузі. При цьому вона надає можливість не тільки зберігати ці знання, а і оперувати ними, наприклад, робити запити щодо дій експерта в тій чи іншій ситуації. Така система може бути використана для допомоги у прийнятті рішень, наприклад, в задачах медичної діагностики без безпосереднього залучення експерта. Існує надзвичайний асортимент експертних систем від мало спеціалізованих діагностичних систем до практично великих і вдосконалених гібридних систем для вирішення складних питань. Експертна система містить базу знань, де зберігається експертна інформація про певну галузь застосування. Окрім бази знань, вона містить механізм виведення для отримання відповідей з огляду на цю інформацію та, можливо, додаткову інформацію про особливості її застосування. Незаповнена база знань разом з механізмом виведення називається оболонкою експертної системи. Оболонку експертної системи підсилюють компонентами для введення інформації в базу знань, і це може бути досягнуто за допомогою клієнтського інтерфейсу для співпраці з експертом, або програмного інтерфейсу для взаємодії з іншими частинами інформаційної системи, що використовує експертну систему як підсистему. Побудова експертної системи передбачає, по-перше, вибір / модифікацію оболонки експертної системи, по-друге, отримання експертної інформації та наповнення інформацією бази знань. Крок другий складніший і потребує більше часу, ніж перший. Існують численні інструменти для створення експертних систем. Сучасні експертні системи є динамічними, тобто такими, що здатні додавати або модифікувати інформацію в базу знань. Експертні системи можуть функціонувати і в

зворотному порядку, тобто не формувати висновки на базі певних фактів, що описують поточну ситуацію, а навпаки, за висновками визначати факти, що призвели до прийняття того чи іншого рішення. Це дозволяє виконати реконструкцію дій людини-оператора, оцінити ефективність його рішень, порівнюючи їх з діями експерта. У експертних системах існує широкий спектр форм подання інформації, найвідомішим є подання на основі правил. Однак ефективність експертної системи в основному залежить від характеру інформації в базі знань експертної системи, а менше - від внутрішнього подання інформації . Це призводить до питання вибору експерта, який сформує базу знань та правила для її модифікації у режимі реального часу.

Як приклад, розглянемо експертну систему CD (Cyber Defense), що вражає великим вибором засобів з інформаційної безпеки, а саме для протидії кібератакам. Вона використовує процедуру порівняння окремого інформаційного процесу з базою знань і у випадку, якщо це відомий безпечний процес, тоді система переходить до перевірки наступного процесу, інакше – інформує про небезпеку. У разі відсутності опису процесу в базі знань експертна система використовує алгоритми виведення знань і оцінює стан системи. Результат такої оцінки – захищений (sheltered), пряма атака (direct) і критичний (extreme). Відповідно до стану машини, експертна система застерігає адміністратора / користувача, а висновок заноситься на базу знань.

Компоненти експертної системи
База знань
Небезпечні IP-адреси
Відоме шкідливе ПЗ
Відомі віруси
Перевірене ПЗ
Перевірені IP-адреси
Результуюча статистика

Механізм виведення
Географічне розташування IP-адреси
Кількість спроб з’єднання
Шаблон з’єднання
Частота використання ПЗ
Частота використання файлу
Час спроби входу в систему
Кількість спроб входу в систему
Порт з’єднання
Шаблон доступу до файлів

Рисунок 2.1 – Компоненти системи експертної безпеки

Нейронні мережі - це один з провідних напрямків ШІ. Нейронні мережі використовують моделі структури людського мозку і його складових - нейронів. Наш розум має велику кількість нейронів, які значною мірою мають загальне призначення і незалежні від області мозку, де вони розташовані. Штучний нейрон (Perceptron) був створений у 1957 році Френком Розенблатом, який заклав основи нейронних систем. Окремий штучний нейрон здатен розв’язувати не дуже складні задачі, але об’єднуючись з іншими нейронами в мережу може відтворювати практично будь-яку з когнітивних властивостей людини. Наприклад, нейромережі навчаються без будь-якої зовнішньої допомоги розпізнавати образи шляхом обробки первинних вхідних даних аналогічно до того, як людський мозок навчається самостійно, використовуючи джерела інформації органів чуття. Нейромережі, що застосовуються в галузі кібербезпеки, можуть визначити, чи є процес або ресурс шкідливим без будь-якого втручання людини. Такий підхід демонструє високоточні результати в детектуванні шкідливого програмного забезпечення, на відміну від класичних методів машинного навчання та розпізнавання образів. Крім того, нейронні мережі дозволяють розпізнати нові небезпеки від шкідливих програм та виявити невідомі вразливості системи.

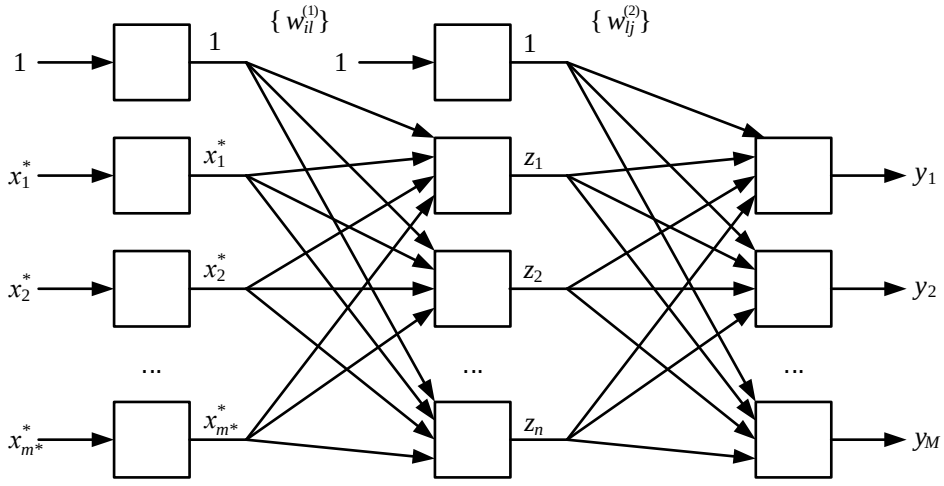

Рисунок 2.2 – Нейромережа

Інтелектуальний агент (ІА) - це автономна сутність, яка контролює процес захисту інформації за допомогою певних сенсорів, а також використовує певні власні виконавчі механізми і механізми координації з іншими ІА для досягнення поставлених цілей. ІА функціонують в режимі реального часу, швидко навчаються новому завдяки можливості безпосередньої взаємодії з навколишнім середовищем та здатні зберігати та використовувати знання. Наприклад, розроблено ряд інтелектуальних агентів для захисту від атак типу DDoS . Якщо ІА, що використовуються для задач інформаційної або кібербезпеки, здатні взаємодіяти між собою, то вони об’єднуються в так звану «Цифрову поліцію» (Digital police).

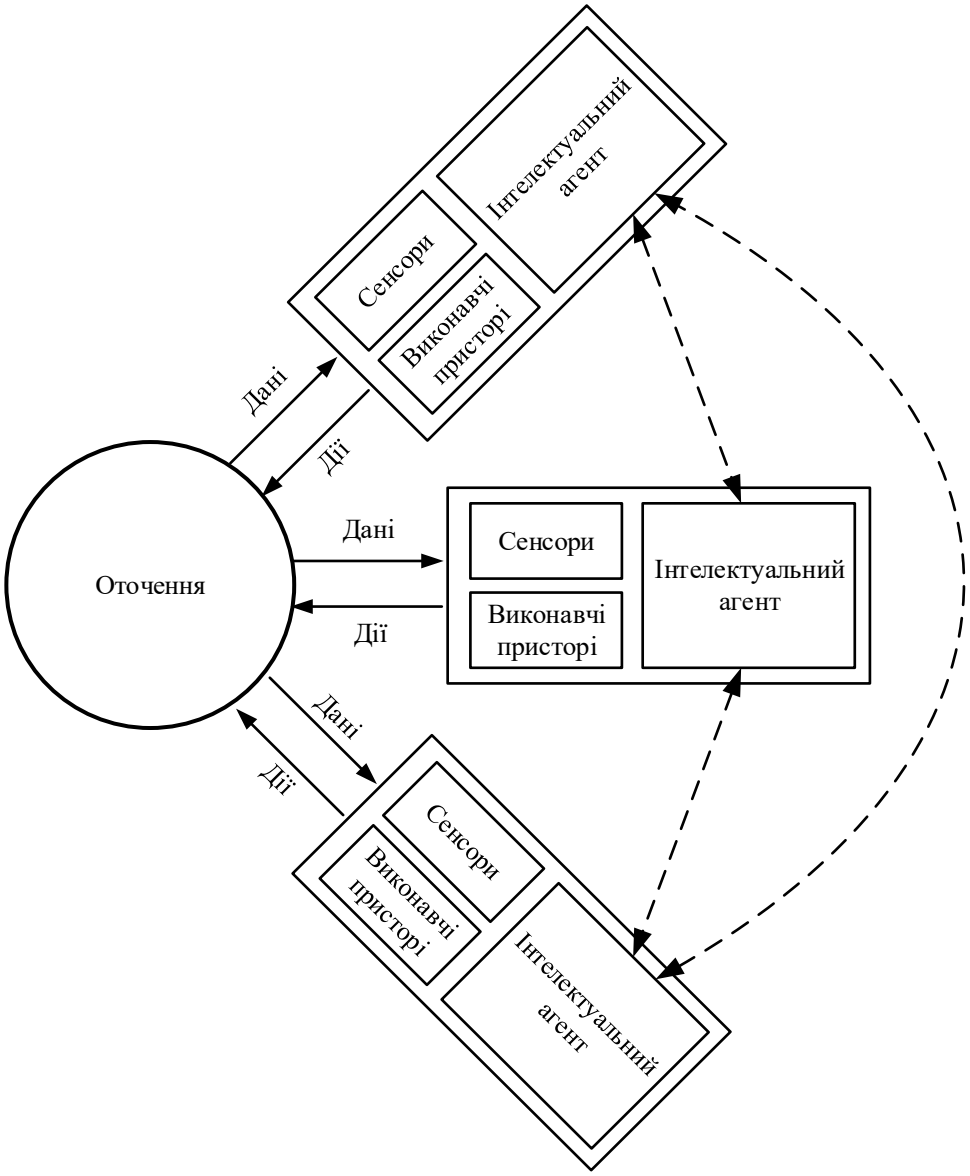

Рисунок 2.3 – Інтелектуальні агенти

2.2 Структура експертних систем

Розробка експертних систем має ряд істотних відмінностей від розробки програмного продукту. Досвід реалізації експертних систем показав, що використання при їх розробці методології, яка прийнята в традиційному програмуванні, або сильно збільшує кількість часу, витраченого на створення експертних систем, або зовсім призводить до негативного результату.

Експертні системи в загальному випадку підрозділяються на статичні і динамічні.

Стандартна статична експертна система складається з таких основних компонентів:

- робочої пам'яті(база даних);
- бази знань;
- інтерпретатор;
- компонентів придбання знань;
- пояснювального компонента;
- діалогового компонента.

Робоча пам'ять призначена для отримання і зберігання вихідних і проміжних даних розв'язуваної в поточний момент завдання.

База знань призначена для зберігання довготривалих даних, що описують конкретну предметну область і правил, що описують раціональне перетворення даних цієї області розв'язуваної задачі.

Інтерпретатор функціонує наступним чином: використовуючи вихідні дані з робочої пам'яті і довготривалі дані з бази знань, він формує правила, застосування яких до вихідних даних призводить до вирішення завдання.

Компонент придбання знань автоматизує процес заповнення експертної системи знаннями експерта. Саме цей компонент забезпечує базу знань всією необхідною інформацією з даної конкретної предметної області.

Компонент пояснень інформує, як система отримала рішення даного завдання, або чому вона це рішення не отримала і які знання вона при цьому

використовувала. Тобто компонент пояснень створює звіт про виконану роботу. Даний компонент є дуже важливим у всій експертній системі, оскільки він значно полегшує тестування системи експертом, а також підвищує довіру користувача до отриманого результату і, отже, прискорює процес розробок.

Діалоговий компонент служить для забезпечення дружнього інтерфейсу користувача як в результаті виконання завдання, так і в процесі набуття знань і оголошення результатів роботи.

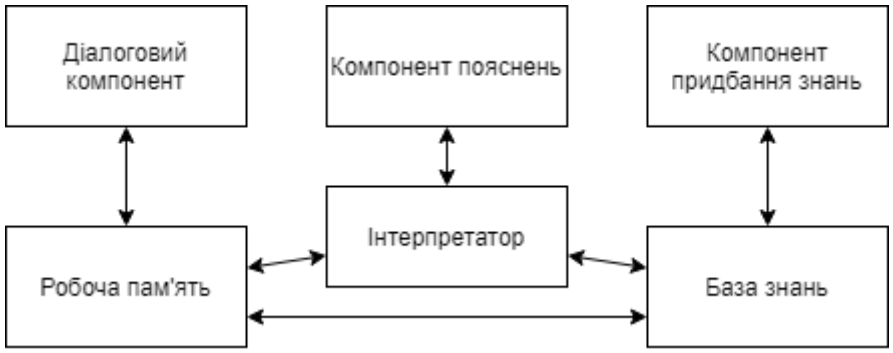

Рисунок 2.4 – Структура ЕС

Статичні експертні системи найчастіше використовуються в технічних додатках, де можна не враховувати зміни навколишнього середовища, що відбуваються під час виконання завдання.

Основні відмінності динамічної експертної системи від статичної.

На відміну від статичної експертної системи в структуру динамічної експертної системи додатково вводяться два наступних компонента:

- підсистема моделювання зовнішнього світу;
- підсистема зв'язків із зовнішнім оточенням.

Підсистема зв'язків із зовнішнім оточенням здійснює зв’язок з зовнішнім світом. Робить вона це за допомогою системи спеціальних датчиків і контролерів.

Крім цього, деякі традиційні компоненти статичної експертної системи піддаються істотним змінам, для того щоб відобразити тимчасову логіку подій, що відбуваються в даний момент в навколишньому середовищі.

Це головна відмінність між статичної та динамічної експертними системами.

Приклад динамічної експертної системи - управління виробництвом різних медикаментів у фармацевтичній промисловості.

Продукційна модель знань.

Продукційні моделі знань близькі до логічних моделей, що дозволяє організувати досить ефективні процедури логічного висновку даних. Якщо розглядати продукційні моделі знань в порівнянні з логічними моделями, то перші більш наочно відображають знання, що є незаперечною перевагою. Тому продукційна модель знань є одним з головних засобів представлення знань в системах штучного інтелекту.

Традиційна продукційна модель знань включає в себе наступні базові компоненти:

- набір правил (або продукцій), що представляють базу знань виробничої системи;
- робочу пам'ять, в якій зберігаються вихідні факти, а також факти, виведені з вихідних фактів за допомогою механізму логічного висновку;
- сам механізм логічного висновку, що дозволяє з наявних фактів, згідно з наявними правилами виведення, виводити нові факти.

Кожне правило, що представляє базу знань виробничої системи, містить умовну і заключну частини. У умовної частини правила знаходиться або одиночний факт, або кілька фактів, з'єднаних кон'юнкцією. У заключній частині правила знаходяться факти, якими необхідно поповнити робочу пам'ять, якщо умовна частина правила є істинною.

Схематичне зображення продукційної моделі знань має такий вигляд:

$$(I)\ Q;\ P;\ A > B;\ N;$$

Тут I - це ім'я продукційної моделі знань або її порядковий номер, за допомогою якого дана продукція виділяється зі всієї безлічі продукційних моделей, отримуючи якусь ідентифікацію. Ім'ям може виступати деяка

лексична одиниця, що відображає суть даної продукції. Фактично продукція називається для кращого сприйняття, щоб спростити пошук потрібної продукції зі списку.

Елемент Q характеризує сферу застосування даної конкретної продукційної моделі знань. Такі сфери легко виділяються в свідомості людини, тому з визначенням даного елемента, як правило, труднощів не виникає.

Розглянемо наступну ситуацію: в однієї сфері нашої свідомості зберігаються знання про те, як треба готувати їжу або як дістатися до роботи, в третій, як правильно експлуатувати пральну машину. Подібне розділення присутнє і пам'яті продукційній моделі знань. Це поділ знань на окремі сфери дозволяє значно економити час, що витрачається на пошук потрібних в даний момент якихось конкретних продукційних моделей знань, і тим самим значно спрощує процес роботи з ними.

Основним елементом продукції є ядро, яке в наведеної вище формулою позначалося як A > B. Ця формула може бути інтерпретована, як «якщо виконується умова A, то слід виконати дію B».

Якщо є більш складна конструкція ядра, то в правій частині допускається наступний альтернативний вибір: «якщо виконується умова A, то слід виконати дію B1, інакше слід виконати дію B2».

Однак інтерпретація ядра продукційної моделі знань може бути різною і залежати від того, що буде стояти ліворуч і праворуч від знаку секвенції «>». При одній з інтерпретацій ядра продукційної моделі знань секвенція може тлумачитися в звичайному логічному сенсі, тобто в якості знаку логічного слідування дії B з істинного умови A.

Проте можливі й інші інтерпретації ядра продукційної моделі знань. Так, наприклад, A може описувати як умова, виконання якої необхідно для того, щоб можна було зробити дію B.

Елемент P визначається, як умова застосовності ядра продукції. Якщо умова P істинно, то ядро продукції активізується. В іншому випадку, якщо умова P не виконується (воно помилкове), ядро не може бути активізовано.

Як наочний приклад розглянемо наступну продукційну модель знань:

«Наявність грошей»; «Якщо хочеш купити річ А, то слід заплатити в касу її вартість і пред'явити чек продавцю».

Якщо умова P істинна, тобто покупка сплачена і чек пред'явлений, то ядро активізується. Покупка здійснена. У разі якщо в цій продукційній моделі знань умова застосовності ядра помилково, тобто якщо немає грошей, то застосувати ядро продукційної моделі знань неможливо, і воно не активізується.

Елемент N називається після-умовою продукційної моделі даних. Після-умова задає дії і процедури, які необхідно виконати після реалізації ядра продукції.

Уявлення знань у вигляді набору правил, тобто за допомогою використання продукційної моделі знань, має такі переваги:

- це простота створення і розуміння окремих правил;
- це простота механізму логічного вибору.

Однак в поданні знань у вигляді набору правил є і недоліки, які обмежують сферу і частоту застосування продукційних моделей знань. Основним таким недоліком вважається неясність взаємних відносин між складовими конкретної продукційної моделі знань правилами, а також правилами логічного вибору.

3 ІНФОРМАЦІЙНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ЕКСПЕРТНОЇ СИСТЕМИ

3.1 Програмні продукти для проектування експертних систем

У розробці експертних систем беруть участь - один або два інженера по знаннях, один експерт і один програміст, який займається модифікацією і узгодженням інструментальних засобів, тобто в розробці бере участь від чотирьох чоловік. При необхідності до процесу розробки експертної системи можуть додаватися й інші учасники.

Для складних систем вважається за доцільне залучати до основного циклу розробки кілька експертів. Однак в такому випадку потрібно, щоб один з експертів відповідав за несуперечливість знань, що повідомляються даними колективом експертів.

Системи і мови програмування, що використовуються для простих програм підходять так само для створення експертних систем, але присутність таких незвичайних для штучного інтелекту складових елементів, як природно-мовний інтерфейс, логічний висновок, робить кращим використання для розробки ЕС таких мов ШІ, як LISP, CLIPS, Prolog і спеціальних засобів підтримки розробки. Основна ідея логічного програмування полягає в відділенні управління ходом обчислень від логіки програми, що робить більш прозорим процес її створення програми. Пролог (Prolog) - мова високого рівня, орієнтований на використання концепцій і методів математичної логіки. Створений він був в Марсельському університеті, що у Франції в 1972 році. Основною його особливістю, що відрізняє Пролог від аналогів, є декларативний характер написаних на ньому програм. Він призначений для розробки програм і систем штучного інтелекту; відноситься до категорії мов 5 покоління. При роботі з ним досить визначити безліч фактів і встановити відносини між ними і тому програмістам не потрібно розписувати крок за кроком процедури. Процедури, вбудовані в мову, отримують логічні висновки

за допомогою цих співвідношень. Ця особливість робить Пролог зручним для написання експертних систем.

Мова Лісп (LISP) розроблений на початку 60-х років в Массачусетському технологічному інституті. Мови програмування Лісп і Пролог мають вбудовані механізми для маніпулювання знаннями. Лісп є універсальною мовою програмування високого рівня і має здатність обробляти спискові структури. Він відноситься до декларативних мов функціонального типу і призначений для обробки символьних даних, представлених у вигляді списків.

Кліпс (CLIPS) був розроблений в середині 80-х років в центрі космічних досліджень NASA. Абревіатура розшифрує як - C Language Integrated Production System. Він включає в мова опису процедур і мова представлення породжують правил. Кліпс містить три основні елементи: блок виведення, базу знань і список фактів, так як використовує продукційну модель подання знань.

Принциповою відмінністю даної системи від інших є те, що вона повністю реалізована на мові «С». Причому вихідні тексти даних програм опубліковані в Інтернеті.

У Кліпс використовується оригінальна LIPS-подібна мова програмування, орієнтований на розробку експертної системи. Крім того, він підтримує ще дві парадигми програмування: процедурну і об'єктно-орієнтовану.

Крім даних трьох мов (Кліпса, Прологу і Ліпса) створено безліч інших мов, орієнтованих на розробку експертних систем і обробку символьної інформації, такі як - Smalltalk, FRL, Interlisp. Так само крім цих спеціалізованих мов для розробки ЕС використовуються і звичайні мови загального призначення - Паскаль, Бейсік, Сі, Асемблер, Фортран, Бейсік і ін.

В програмних інструментальних засобів виділяють такі групи

- Символьні мови програмування (LISP, INTERLISP, SMALLTALK, CLIPS);

- Мови інженерії знань, тобто мови програмування, що дозволяють реалізувати один із способів подання знань (OPS5, LOOPS, KES, Prolog);
- Оболонки експертних систем (або порожні експертні системи), тобто системи, що не містять знань ні про яку предметної області (EMYCIN, ЕКО, ЕКСПЕРТ, EXSYS RuleBook, Expert System Creator).

При розробці ЕС використовується концепція "швидкого прототипу". Сенс «швидкого прототипу» в тому, що розробники не створюють відразу кінцевий продукт. На початковому етапі розробники створюють прототип ЕС. Вони повинні відповідати двом вимогам. З однієї сторони, вони повинні вирішувати типові завдання конкретного додатка. З іншої сторони - час і трудомісткість розробки мають бути незначні, для того щоб була можливість максимально розпаралелити процес накопичення і налагодження знань з процесом розробки програмних засобів. При створенні прототипу використовуються різні засоби, які пришвидшують процес проектування для задоволення зазначеним вимогам.

Прототип повинен відобразити придатність методів інженерії знань для цього додатка. У разі успішного результату експерт та інженер по знаннях розширюють знання прототипу про проблемну область. В іншому випадку можливо потрібне проектування нового прототипу або розробники можуть прийти до ідеї про непридатність методів експертних систем для цього додатку. Під час накопичення знань прототип може досягти стану, коли він успішно виконує всі завдання програми. Зміна прототипу ЕС в кінцевий продукт зазвичай призводить до його перепрограмування на мовах низького рівня, що забезпечують як збільшення швидкодії, так і економії необхідної пам'яті. Час створення і трудомісткість ЕС залежать від типу застосовуваного інструментарію.

Поділ інструментальних засобів розробки експертних систем, як правило, відбувається за такими параметрами: механізми виведення і моделювання; приклади програмування та механізми реалізації; рівень

використовуваної мови; спосіб представлення знань; кошти отримання знань; технології розробки.

Рівень використовуваного мови:

- традиційні (в тому числі і об'єктно-орієнтовані) мови програмування;
- спеціальні мови програмування (LISP, PROLOG, РЕФАЛ);
- інструментальні засоби, що містять частину компонентів експертних систем (призначені для розробників ЕС);
- середовища розробки загального призначення, що містять всі елементи експертних систем, але не мають описи приватних проблемних середовищ;
- проблемно-орієнтовані середовища розробки (для вирішення певного класу задач або мають знання про типи предметних областей).

Парадигми програмування:

- процедурне програмування;
- програмування, орієнтоване на дані;
- програмування, орієнтоване на правила;
- об'єктно-орієнтоване програмування;
- логічне програмування.

Спосіб (моделі) подання знань: фрейми (об'єкти); продукційні правила; семантичні мережі; логічні формули; нейронні мережі. В результаті аналізу засобів проектування і розробки ЕС зроблені наступні висновки. Розробка ЕС має немалі відмінності від розробки звичайного програмного продукту. Використання при розробці методології, прийнятої в традиційному програмуванні, або уповільнює процес створення ЕС, або призводить до негативного результату.

Загальним недоліком мов програмування для створення експертних систем є: великий час розробки готової системи, необхідність залучення висококваліфікованих програмістів, труднощі з модифікацією готової системи.

3.2 Формування бази знань

Формуючи базу знань треба враховувати, що будь-яка комп’ютерна система це складний, багато модульний механізм. При оцінці рівня її безпеки можлива ситуація при якій однакова симптоматика викликається різними логічними правилами. Так само часто мають місце некоректні або завідомо неправдиві уявлення про систему користувачем. При створенні експертної системи робимо пропущення, що користувач може ввести дані, які потенційно приведуть до помилкової оцінки захищеності системи або зовсім відсутність оцінки, так як оцінювання відбувається відносно нормативних документів, в яких конкретно описаний кожен крок.

Організаційно-технічна система, навколишнє середовище(фізичне), інформація яка обробляється і персонал є складовими автоматизованої системи. Від цих параметрів залежать вимоги до функціонального складу комплексу засобів захисту. Вимоги гарантій залежать від виду оброблювальної інформації та призначенням автоматизованої системи.

Відповідно до цих вимог можна виділити три класи автоматизованих систем.

Клас «1» - комплекс с однією комп’ютерною системою та одним користувачем, що працює з інформацію кількох(або однієї) ступенів обмеження доступу.

- З комплексом дозволено працювати одному користувачу, проте у загальному випадку осіб, що мають доступ до комплексу, може бути декілька.
- Користувачі можуть мати права доступу до оброблювальної інформації.

Клас «2» — локальна система з користувачами та комп’ютерними системами в якій може оброблятися інформація різного типу приватності.

Клас другого типу відрізняється наявністю користувачів з різними правами доступу або технічних засобів.

Клас «3» — розподілена система, яка складається с багатьох комп’ютерних систем та користувачів.

Кожен клас автоматизованої системи має класифікацію, яка базується на вимогах до забезпечення безпеки властивостей інформації. Інформація має три параметри: конфіденційність, цілісність і доступність. Наслідком цього є виділення в кожному класі таких підкласів згідно з НД ТЗІ 2.5-005 -99[8]:

- автоматизована система, в якій підвищені вимоги до — забезпечення конфіденційності оброблюваної інформації;
- автоматизована система, в якій підвищені вимоги до — забезпечення цілісності оброблюваної інформації;
- автоматизована система, в якій підвищені вимоги до — забезпечення доступності оброблюваної інформації;
- автоматизована система, в якій підвищені вимоги до — забезпечення конфіденційності і цілісності оброблюваної інформації;
- автоматизована система, в якій підвищені вимоги до — забезпечення конфіденційності і доступності оброблюваної інформації;
- автоматизована система, в якій підвищені вимоги до — забезпечення цілісності і доступності оброблюваної інформації.
- автоматизована система, в якій підвищені вимоги до — забезпечення конфіденційності, цілісності і доступності оброблюваної інформації.

Для кожного підкласу автоматизованої системи інтегруються стандартні функціональні профілі(ієрархічні). Реалізація профілів надає прогресуючу захищеність стосовно конфіденційності, цілісності і доступності.

Основою для стандартних функціональних профілів є реальні вимоги захисту оброблювальної інформації від загроз та функціональні послуги, які дають можливість підтримувати рівень безпеки, встановлений політикою безпеки.

В комп’ютерній системі політика безпеки, в якому створено стандартний профіль захисту повинен базуватися на відповідних документах, що мають вимого щодо обробки інформації в автоматизованій системі.

Для стандартних функціональних профілів є можливість не реалізовувати зв’язаної з ними політики безпеки або рівня гарантій.

Єдина вимога, якої слід дотримуватися при утворенні нових профілів, — це додержання описаних в НД ТЗІ 2.5-004-99 «Критерії оцінки захищеності інформації в комп’ютерних системах від несанкціонованого доступу» необхідних умов для кожної із послуг, що включаються до профілю.

3.2.1 Стандартні функціональні профілі захищеності

Цитуючи документ «НД ТЗІ 2.5-005 -99» приведемо класифікацію профілів захищеності.

«Стандартні функціональні профілі захищеності для автоматизованих систем класу 1

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 1, головною вимогою до яких є забезпечення конфіденційності оброблюваної інформації:

1.К.1 = { НР-1, НИ-1, НК-1, НО-1, НЦ-1, НТ-1 }

1.К.2 = { КА-1, КО-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-1 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 1, головною вимогою до яких є забезпечення цілісності оброблюваної інформації :

1.Ц.1 = { НР-1, НИ-1, НК-1, НО-1, НЦ-1, НТ-1 }

1.Ц.2 = { ЦА-2, ЦО-1,НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-1 }

Стандарті функціональні профілі захищеності в КС, що входять до складу АС класу 1, головною вимогою до яких є забезпечення доступності оброблюваної інформації:

1.Д.1 = { ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-1 }

1.Д.2 = { ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-2 }

1.Д.3 = { ДР-2, ДС-2, ДЗ-2, ДВ-2, НР-3, НИ-2, НК-1, НО-1, НЦ-2, НТ-2 }

1.Д.4 = { ДР-2, ДС-3, ДЗ-3, ДВ-3,

НР-4, НИ-2, НК-1, НО-1, НЦ-2, НТ-2 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 1, з підвищеними вимогами до забезпечення конфіденційності і цілісності оброблюваної інформації:

1.КЦ.1 = { НР-1, НИ-1, НК-1, НО-1, НЦ-1, НТ-1 }

1.КЦ.2 = { КА-1, КО-1, ЦА-2, ЦО-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-1 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 1, з підвищеними вимогами до забезпечення конфіденційності і доступності оброблюваної інформації;

1.КД.1 = { КА-1, КО-1, ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-1 }

1.КД.2 = { КА-1, КО-1, ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-2 }

1.КД.3 = { КА-1, КО-1, ДР-2, ДС-2, ДЗ-2, ДВ-2, НР-3, НИ-2, НК-1, НО-1, НЦ-2, НТ-2 }

1.КД.4 = { КА-1, КО-1, ДР-2, ДС-3, ДЗ-3, ДВ-3, НР-4, НИ-2, НК-1, НО-1, НЦ-2, НТ-2 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 1, з підвищеними вимогами до забезпечення цілісності і доступності інформації:

1.ЦД.1 = { ЦА-1, ЦО-1, ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-1 }

1.ЦД.2 = { ЦА-1, ЦО-1, ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-2 }

1.ЦД.3 = { ЦА-1, ЦО-1, ДР-2, ДС-2, ДЗ-2, ДВ-2, НР-3, НИ-2, НК-1, НО-1, НЦ-2, НТ-2 }

1.ЦД.4 = { ЦА-1, ЦО-1, ДР-2, ДС-3, ДЗ-3, ДВ-3, НР-4, НИ-2, НК-1, НО-1, НЦ-2, НТ-2 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 1, з підвищеними вимогами до забезпечення конфіденційності, цілісності і доступності інформації:

1.КЦД.1 = { КА-1, КО-1, ЦА-1, ЦО-1, ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-1 }

1.КЦД.2 = { КА-1, КО-1, ЦА-1, ЦО-1, ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-2 }

1.КЦД.3 = { КА-1, КО-1, ЦА-1, ЦО-1, ДР-2, ДС-2, ДЗ-2, ДВ-2, НР-3, НИ-2, НК-1, НО-1, НЦ-2, НТ-2 }

1.КЦД.4 = { КА-1, КО-1, ЦА-1, ЦО-1, ДР-2, ДС-3, ДЗ-3, ДВ-3, НР-4, НИ-2, НК-1, НО-1, НЦ-2, НТ-2 }

Стандартні функціональні профілі для автоматизованих систем класу 2

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 2, головною вимогою до яких є забезпечення конфіденційності оброблюваної інформації:

2.К.1 = { КД-2, НР-2, НИ-2, НК-1, НО-1, НЦ-1 }

2.К.2 = { КД-2, КО-1, НР-2, НИ-2, НК-1, НО-1, НЦ-2, НТ-1 }

2.К.3 = { КД-2, КА-2, КО-1, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2 }

2.К.4 = { КД-2, КА-2, КО-1, КК-1, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2 }

2.К.5 = { КД-3, КА-3, КО-1, КК-1, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2 }

2.К.6 = { КД-4, КА-4, КО-1, КК-2, НР-5, НИ-2, НК-2, НО-3, НЦ-3, НТ-2 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 2, головною вимогою до яких є забезпечення цілісності інформації:

2.Ц.1 = { ЦД-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1 }

2.Ц.2 = { ЦД-1, ЦО-1, НР-2, НИ-2, НК-1, НО-1, НЦ-2, НТ-1 }

2.Ц.3 = { ЦД-1, ЦА-2, ЦО-1, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2 }

2.Ц.4 = { КО-1, ЦД-1, ЦА-3, ЦО-2, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2 }

2.Ц.5 = { КО-1, ЦД-4, ЦА-4, ЦО-2, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2 }

Стандарті функціональні профілі захищеності в КС, що входять до складу АС класу 2, головною вимогою до яких є забезпечення доступності оброблюваної інформації:

2.Д.1 = { ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-1 }

2.Д.2 = { ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-2 }

2.Д.3 = { ДР-3, ДС-2, ДЗ-2, ДВ-2, НР-3, НИ-2, НК-1, НО-1, НЦ-2, НТ-2 }

2.Д.4 = { ДР-3, ДС-3, ДЗ-3, ДВ-3, НР-4, НИ-2, НК-1, НО-1, НЦ-2, НТ-2 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 2, з підвищеними вимогами до забезпечення конфіденційності і цілісності оброблюваної інформації:

2.КЦ.1 = { КД-2, ЦД-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1 }

2.КЦ.2 = { КД-2, КО-1, ЦД-1, ЦО-1, НР-2, НИ-2, НК-1, НО-1, НЦ-2, НТ-1 }

2.КЦ.3 = { КД-2, КА-2, КО-1, ЦД-1, ЦА-2, ЦО-1, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2 }

2.КЦ.4 = { КД-2, КА-2, КО-1, КК-1, ЦД-1, ЦА-2, ЦО-1, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2 }

2.КЦ.5 = { КД-3, КА-3, КО-1, КК-1, ЦД-1, ЦА-3, ЦО-2, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2 }

2.КЦ.6 = { КД-4, КА-4, КО-1, КК-2, ЦД-4, ЦА-4, ЦО-2, НР-5, НИ-2, НК-2, НО-3, НЦ-3, НТ-2 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 2, з підвищеними вимогами до забезпечення конфіденційності і доступності оброблюваної інформації:

2.КД.1 = { КД-2, КА-2, КО-1, ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2 }

2.КД.2 = { КД-2, КА-2, КО-1, КК-1, ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2 }

2.КД.3 = { КД-3, КА-3, КО-1, КК-1, ДР-3, ДС-2, ДЗ-2, ДВ-2, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2 }

2.КД.4 = { КД-4, КА-4, КО-1, КК-2, ДР-3, ДС-3, ДЗ-3, ДВ-3, НР-5, НИ-2, НК-2, НО-3, НЦ-3, НТ-2 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 2, з підвищеними вимогами до забезпечення цілісності і доступності оброблюваної інформації:

2.ЦД.1 = { ЦД-1, ЦО-1, ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-2, НТ-1 }

2.ЦД.2 = { ЦД-1, ЦА-2, ЦО-1, ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2 }

2.ЦД.3 = { КО-1, ЦД-1, ЦА-3, ЦО-2, ДР-3, ДС-2, ДЗ-2, ДВ-2, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2 }

2.ЦД.4 = { КО-1, ЦД-4, ЦА-4, ЦО-2, ДР-3, ДС-3, ДЗ-3, ДВ-3, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 2, з підвищеними вимогами до забезпечення конфіденційності, цілісності і доступності оброблюваної інформації:

2.КЦД.1 = { КД-2, КО-1, ЦД-1, ЦО-1, ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2 }

2.КЦД.2 = { КД-2, КА-2, КО-1, ЦД-1, ЦА-2, ЦО-1, ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2 }

2.КЦД.3 = { КД-2, КА-2, КО-1, КК-1, ЦД-1, ЦА-3, ЦО-2, ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2 }

2.КЦД.4 = { КД-3, КА-3, КО-1, КК-1, ЦД-1, ЦА-3, ЦО-2, ДР-3, ДС-2, ДЗ-2, ДВ-2, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2 }

2.КЦД.5 = { КД-4, КА-4, КО-1, КК-2, ЦД-4, ЦА-4, ЦО-2, ДР-3, ДС-3, ДЗ-3, ДВ-3, НР-5, НИ-2, НК-2, НО-3, НЦ-3, НТ-2 }

Стандартні функціональні профілі захищеності для автоматизованих систем класу 3

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 3, головною вимогою до яких є забезпечення конфіденційності оброблюваної інформації:

3.К.1 = { КД-2, КВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НВ-1 }

3.К.2 = { КД-2, КО-1, КВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-2, НТ-1, НВ-1 }

3.К.3 = { КД-2, КА-2, КО-1, КВ-2, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2, НВ-1 }

3.К.4 = { КД-2, КА-2, КО-1, КК-1, КВ-3, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2, НВ-2 }

3.К.5 = { КД-3, КА-3, КО-1, КК-1, КВ-4, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2, НВ-2 }

3.К.6 = { КД-4, КА-4, КО-1, КК-2, КВ-4, НР-5, НИ-2, НК-2, НО-3, НЦ-3, НТ-2, НВ-2, НА-1 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 3, головною вимогою до яких є забезпечення цілісності оброблюваної інформації:

3.Ц.1 = { ЦД-1, ЦВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НВ-1 }

3.Ц.2 = { ЦД-1, ЦО-1, ЦВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-2, НТ-1, НВ-1 }

3.Ц.3 = { ЦД-1, ЦА-2, ЦО-1, ЦВ-2, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2, НВ-2, НА-1 }

3.Ц.4 = { КО-1, ЦД-1, ЦА-3, ЦО-2, ЦВ-2, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2, НВ-2, НА-1, НП-1 }

3.Ц.5 = { КО-1, ЦД-4, ЦА-4, ЦО-2, ЦВ-3, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2, НВ-3, НА-2, НП-2 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 3, головною вимогою до яких є забезпечення доступності оброблюваної інформації:

3.Д.1 = { ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-1, НВ-1 }

3.Д.2 = { ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НТ-2, НВ-1 }

3.Д.3 = { ДР-3, ДС-2, ДЗ-2, ДВ-2, НР-3, НИ-2, НК-1, НО-1, НЦ-2, НТ-2, НВ-1 }

3.Д.4 = { ДР-3, ДС-3, ДЗ-3, ДВ-3, НР-4, НИ-2, НК-1, НО-1, НЦ-2, НТ-2, НВ-1 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 3, з підвищеними вимогами до забезпечення конфіденційності і цілісності оброблюваної інформації:

3.КЦ.1 = { КД-2, КВ-1, ЦД-1, ЦВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-1, НВ-1 }

3.КЦ.2 = { КД-2, КО-1, КВ-1, ЦД-1, ЦО-1, ЦВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-2, НТ-1, НВ-1 }

3.КЦ.3 = { КД-2, КА-2, КО-1, КВ-2, ЦД-1, ЦА-2, ЦО-1, ЦВ-2, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2, НВ-1 }

3.КЦ.4 = { КД-2, КА-2, КО-1, КК-1, КВ-3, ЦД-1, ЦА-2, ЦО-1, ЦВ-2, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2, НВ-2 }

3.КЦ.5 = { КД-3, КА-3, КО-1, КК-1, КВ-3, ЦД-1, ЦА-3, ЦО-2, ЦВ-2, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2, НВ-2, НА-1, НП-1 }

3.КЦ.6 = { КД-4, КА-4, КО-1, КК-2, КВ-4, ЦД-4, ЦА-4, ЦО-2, ЦВ-3, НР-5, НИ-2, НК-2, НО-3, НЦ-3, НТ-2, НВ-2, НА-1, НП-1 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 3, з підвищеними вимогами до забезпечення конфіденційності і доступності оброблюваної інформації:

3.КД.1 = { КД-2, КА-2, КО-1, КВ-2, ДР-1, ДВ-1,НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2, НВ-1 }

3.КД.2 = { КД-2, КА-2, КО-1, КК-1, КВ-3, ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2, НВ-2 }

3.КД.3 = { КД-3, КА-3, КО-1, КК-1, КВ-4, ДР-3, ДС-2, ДЗ-2, ДВ-2, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2, НВ-2 }

3.КД.4 = { КД-4, КА-4, КО-1, КК-2, КВ-4, ДР-3, ДС-3, ДЗ-3, ДВ-3, НР-5, НИ-2, НК-2, НО-3, НЦ-3, НТ-2, НВ-2, НА-1 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 3, з підвищеними вимогами до забезпечення цілісності і доступності оброблюваної інформації:

3.ЦД.1 = { ЦД-1, ЦО-1, ЦВ-1, ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-1, НЦ-2, НТ-1, НВ-1 }

3.ЦД.2 = { ЦД-1, ЦА-2, ЦО-1, ЦВ-2, ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2, НВ-2, НА-1 }

3.ЦД.3 = { КО-1, ЦД-1, ЦА-3, ЦО-2, ЦВ-2, ДР-3, ДС-2, ДЗ-2, ДВ-2, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2, НВ-2, НА-1, НП-1 }

3.ЦД.4 = { КО-1, ЦД-4, ЦА-4, ЦО-2, ЦВ-3, ДР-3, ДС-3, ДЗ-3, ДВ-3, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2, НВ-3, НА-2, НП-2 }

Стандартні функціональні профілі захищеності в КС, що входять до складу АС класу 3, з підвищеними вимогами до забезпечення конфіденційності, цілісності і доступності оброблюваної інформації:

3.КЦД.1 = { КД-2, КО-1, КВ-1, ЦД-1, ЦО-1, ЦВ-1, ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2, НВ-1 }

3.КЦД.2 = { КД-2, КА-2, КО-1, КВ-2, ЦД-1, ЦА-2, ЦО-1, ЦВ-2, ДР-1, ДВ-1, НР-2, НИ-2, НК-1, НО-2, НЦ-2, НТ-2, НВ-1 }

3.КЦД.3 = { КД-2, КА-2, КО-1, КК-1, КВ-3, ЦД-1, ЦА-3, ЦО-2, ЦВ-2, ДР-2, ДС-1, ДЗ-1, ДВ-2, НР-3, НИ-2, НК-1, НО-2, НЦ-3, НТ-2, НВ-2 }

3.КЦД.4 = { КД-3, КА-3, КО-1, КК-1, КВ-3, ЦД-1, ЦА-3, ЦО-2, ЦВ-2, ДР-3, ДС-2, ДЗ-2, ДВ-2, НР-4, НИ-2, НК-1, НО-3, НЦ-3, НТ-2, НВ-2, НА-1, НП-1 }

3.КЦД.5 = { КД-4, КА-4, КО-1, КК-2, КВ-4, ЦД-4, ЦА-4, ЦО-2, ЦВ-3, ДР-3, ДС-3, ДЗ-3, ДВ-3, НР-5, НИ-2, НК-2, НО-3, НЦ-3, НТ-2, НА-1, НП-1, НВ-2, НА-1, НП-1 }»[8].

3.3 Програмна реалізація експертної системи

Програмна реалізація експертної системи оцінки захищеності автоматизованої системи виконана за допомогою мови CLIPS. CLIPS - це сучасний інструмент для створення експертних систем написаний на мові С. Компоненти CLIPS:

- інтерактивне середовище;
- гнучка і потужна мови ;
- декілька допоміжних інструментів.

CLIPS має ліцензію, що дозволяє поширювати та використовувати програмне забезпечення безкоштовно. Це стало причиною до популяції даного ПЗ(створення бібліотек, вдосконалення архітектури користувачами і т.п.). У роботі CLIPS версії 6.34. Для комфортної роботи експертної системи користувач замість ручного введення всіх фактів, які відображають несправності буде працювати з системою «опитування». Був обраний саме цей метод, бо користувач не може наперед знати, як само треба описати свою проблему або не ввів факти, які були вважливі для рішення експертної системи. Для того, щоб не було помилок опису інформації була і реалізована продукційна модель експертної системи.

При моделюванні продукцій(правил) було вирішено створити систему, яка буде задавати користувачу питання відповідно до етапу роботи ЕС і кожен наступний крок буде проводитись враховуючи попередню відповідь. Для цього формат відповіді повинен бути строгим, що було реалізовано за допомогою вводу відповідей із запропонованих варіантів відповіді або питання, відповіді на які можуть бути тільки «так» чи «ні».

Реалізація однієї з функції наведена нижче.

```
(deffunction ask-question (?question $?allowed-values)
(printout t ?question)
(bind ?answer (read))
(if (lexemep ?answer)
then (bind ?answer (lowcase ?answer)))
(while (not (member ?answer ?allowed-values)) do
(printout t ?question)
(bind ?answer (read))
(if (lexemep ?answer)
then (bind ?answer (lowcase ?answer))))
?answer)
```

Рисунок 3.1 Функція запитання

Параметрами функції є змінна question, яка зберігає повідомлення запитання і складова змінна allowed-values, яка призначена для зберігання коректних даних відповіді на поточне питання.

Після виклику функція виводить відповідне питання і зчитує відповідь в змінну answer. Якщо змінна answer містить текст, то вона буде примусово приведена до прописного алфавіту. Потім слідує перевірка чи отримана відповідь є однією із заданих(allowed-values). Процес запиту повторюється циклічно поки зчитувана змінна не буде коректною. Якщо процес зчитування успішний, то виведеться повідомлення.

Реалізація функції, яка буде викликатися при запитанні, відповідь на яке може бути тільки «так» чи «ні».

```
(deffunction yes-or-no-p (?question)
(bind ?response (ask-question ?question так ні т н))
(if (or (eq ?response так) (eq ?response т))
then TRUE
else FALSE))
```

Рисунок 3.2 Функція так-ні

Функція yes-or-no-p викликає функцію ask-question с параметрами які є статичними. При вводі користувачем відповіді так чи ні, функція повертає значення TRUE, інакше FALSE.

```
(defrule as-class
(as-class) =>
(format t "1 - single-machine single-user system%n2 - local multi-machines
multi-users system%n3 - distributed multi-machines multi-users system%n")
(bind ?response (ask-question "What is the type of system: " 1 2 3))
(retract 2)
(if (eq ?response 1)
  then
  (assert (as-class 1))
  else
   (if (eq ?response 2)
    then (assert (as-class 2))
    else
     (if (eq ?response 3)
      then (assert (as-class 3))
      else (assert (as-class))))))
```

Рисунок 3.3 Правило для класу АС

Умовний елемент (as-class) вказує, що певної інформації про клас системи ще немає і подібний запит буде заданий користувачеві першим. Інші правила бази знань містять певний набір умов їх застосування. Наприклад, необхідними умовами початку оцінки критерія спостереженості для автоматизованої системи класу 1, а саме реєстрація, буде наявність факту, що система відноситься до класу 1 (as-class 1) та відсутність рівня цього критерія

```
(defrule as-class-1-NR
    (as-class 1)
    (not (NR))=>
    (format t "1 - безпосереднє відношення до безпеки%n2 - безпосереднє або непряме відношення
    до безпеки%n")
    (bind ?response-1 (ask-question "КЗЗ здатна здійснювати реєстрацію подій, що мають (1 або
    2): " 1 2))
    (if (eq ?response-1 1)
        then
            (format t "1 - КЗЗ здатна передавати журнал реєстрації в інші системи з
            використанням певних механізмів захисту%n2 - КЗЗ забезпечує захист журналу
            реєстрації від несанкціонованого доступу, модифікації або руйнування%n")
            (bind ?response-2 (ask-question "Оберіть одне з двух тверджень: " 1 2))
            (if (eq ?response-2 1)
                then
                    (assert (NR 1))
                else
                    (if (yes-or-no-p "КЗЗ здатна контролювати одиничні або повторювані
                    реєстраційні події?(так або ні): ")
                        then
                            (assert (NR 3))
                        else
                            (assert (NR 2))
                    )
            )
        else
                (if (yes-or-no-p "КЗЗ здатна виявляти і аналізувати несанкціоновані дії в
                реальному часі?(так або ні): ")
                    then
                        (assert (NR 5))
                    else
                        (assert (NR 4))
                )
    )
)
```

Рисунок 3.4 Правило для класу НР

Для того, щоб запустити експертну систему, досить виконати команду reset, яка додасть факт initial-fact, необхідний для правила system-banner, і команду run. Після цього користувач побачить повідомлення " System security assessment system", яке означає те, що система почала працювати, і отримає серію питань, відповіді на які допоможуть експертній системі оцінити стан захищеності системи і підібрати профіль відповідно до вимог, які були описані раніше.

3.4 Тестування експертної системи

Працездатність розробленої експертної системи перевіряється шляхом розв'язування задачі оцінки КС. Для демонстрації роботи ЕС запустимо програму командою (run) в середовищі розробки CLIPS.

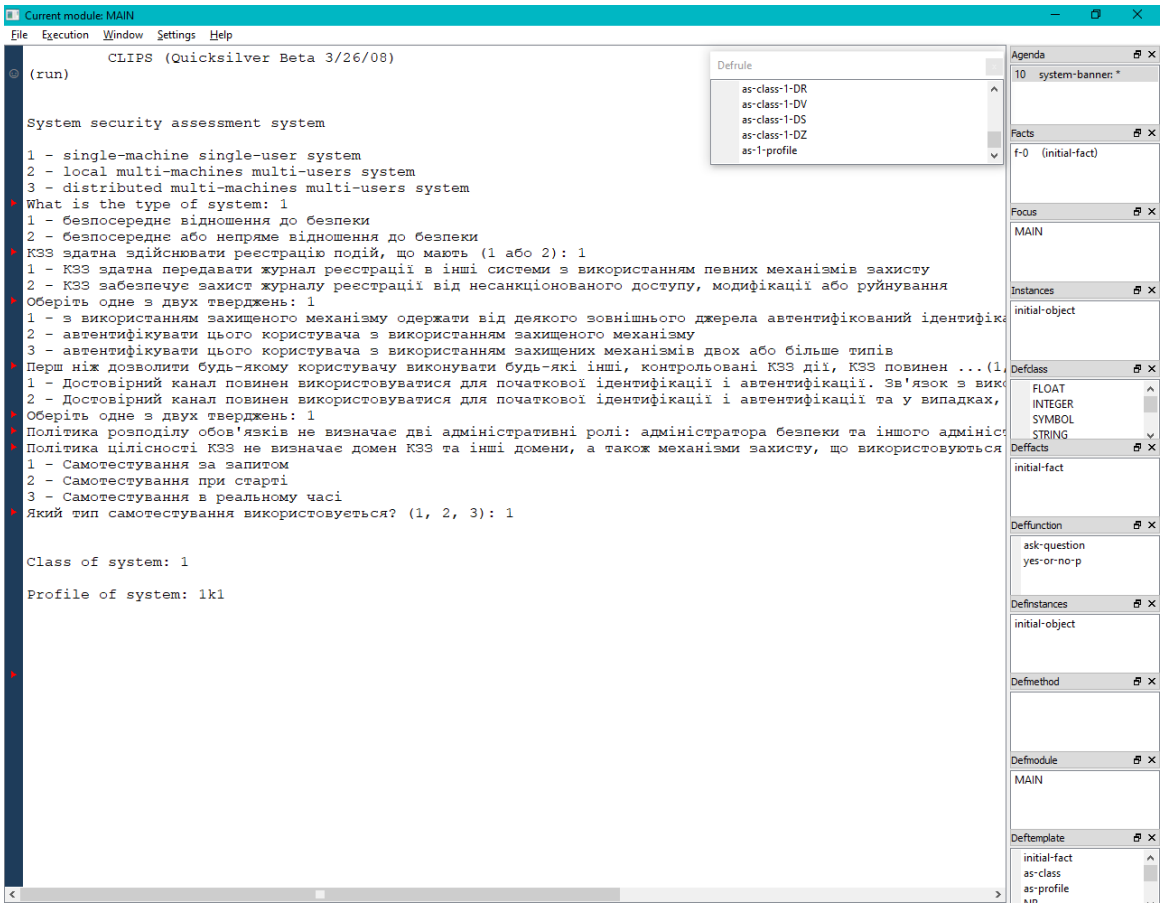

Рисунок 3.5 Тестування ЕС

Процес роботи с ЕС відбувається в режимі опитування. Користувач, який використовує дану систему відповідає на запитання, внаслідок чого в системі створюються певні факти, на базі яких формується висновок роботи

програми. В даному випадку після того як користувач завершив роботу з ЕС, він отримав клас оцінки захищеності його системи(1к1). Згідно стандарту значення 1к1 відповідає стандартному функціональному профілю захищеності в КС, що входять до складу АС класу 1, головною вимогою до яких є забезпечення конфіденційності оброблюваної інформації.

ВИСНОВОК

У випускній роботі була спроектована і реалізована експертна система оцінки захищеності КС від несанкціонованого доступу з використанням продукційної моделі подання знань. Проведений аналітичний огляд різних методів забезпеченню безпеки КС. Описана математична модель експертних систем. Сформована база знань складалася з 16 правил типу ЯКЩО-ТО, для зберігання і обробки яких застосовувалось середовище для проектування експертних систем CLIPS. Працездатність експертної системи перевірялася шляхом оцінки комп’ютерної системи на захищеність. Далі планується з метою підвищення ефективності розробленої експертної системи розширити базу знань, а також провести візуалізацію процесу оцінювання з використанням графічних матеріалів.

СПИСОК ЛІТЕРАТУРИ

1. Bramer M. Logic Programming with Prolog. – Springer, 2014. — 253 p.
2. Nalepa G. Modeling with Rules Using Semantic Knowledge Engineering. – Springer, 2018. — 453 p.
3. Galar D., Kumar U. eMaintenance: Essential Electronic Tools for Efficiency.– Academic Press, 2017. — 549 p.
4. Ryan Darrel (ed.). Expert Systems: Design, Applications and Technology.– Nova Science Publishers, 2017. — 149 p
5. Biffl Stefan, Sabou Marta. Semantic Web Technologies for Intelligent Engineering Applications. – Springer, 2016. — 413 p.
6. Burgin Mark. Theory Of Knowledge: Structures And Processes. – World Scientific, 2016. — 957 p.
7. Yurin A.Y., Dorodnykh N. O. Personal knowledge base designer: Software for expert systems prototyping .– SoftwareX, 2020.– Vol. 11, Article number: 100411
8. Класифікація автоматизованих систем і стандартні функціональні профілі захищеності оброблюваної інформації від несанкціонованого доступу.– http://www.dsszzi.gov.ua/dsszzi/control/uk/publish/article?showHidden=1&art_id=101870&cat_id=89734&ctime=1344501089407
9. Искусственный интеллект [Електронний ресурс] // – Режим доступу: https://www.twirpx.com/files/science/informatics/ai/
10. Безопаность в КС [Електронний ресурс] // – Режим доступу: https://www.twirpx.com/files/science/informatics/security/
11. Introducing Libgen [Електронний ресурс] // – Режим доступу: http://gen.lib.rus.ec/search.php?&req=topicid77&phrase=0&view=simple&column=topic&sort=year&sortmode=DESC

ДОДАТОК

```
(defrule as-class-1-NR
    (as-class 1)
    (not (NR))=>
    (format t "1 - безпосереднє відношення до безпеки%n2 - безпосереднє або непряме відношення до
    безпеки%n")
    (bind ?response-1 (ask-question "КЗЗ здатна здійснювати реєстрацію подій, що мають (1 або 2): "
    1 2))
    (if (eq ?response-1 1)
        then
            (format t "1 - КЗЗ здатна передавати журнал реєстрації в інші системи з використанням
            певних механізмів захисту%n2 - КЗЗ забезпечує захист журналу реєстрації від
            несанкціонованого доступу, модифікації або руйнування%n")
            (bind ?response-2 (ask-question "Оберіть одне з двух тверджень: " 1 2))
            (if (eq ?response-2 1)
                then
                    (assert (NR 1))
                else
                    (if (yes-or-no-p "КЗЗ здатна контролювати одиничні або повторювані реєстраційні
                    події?(так або ні): ")
                        then
                            (assert (NR 3))
                        else
                            (assert (NR 2))
                    )
            )
        else
                (if (yes-or-no-p "КЗЗ здатна виявляти і аналізувати несанкціоновані дії в реальному
                часі?(так або ні): ")
                    then
                        (assert (NR 5))
                    else
                        (assert (NR 4))
                )
    )
)
```

```
(defrule as-class-1-NU
    (as-class 1)
    (not (NU))=>
    (format t "1 - з використанням захищеного механізму одержати від деякого зовнішнього джерела
    автентифікований ідентифікатор цього користувача%n")
    (format t "2 - автентифікувати цього користувача з використанням захищеного механізму%n")
    (format t "3 - автентифікувати цього користувача з використанням захищених механізмів двох або
    більше типів%n")
    (bind ?response-1 (ask-question "Перш ніж дозволити будь-якому користувачу виконувати будь-які
    інші, контрольовані КЗЗ дії, КЗЗ повинен ...(1, 2, 3): " 1 2 3))
    (if (eq ?response-1 1)
        then
            (assert (NU 1))
        else
                (if (eq ?response-1 2)
                    then
                        (assert (NU 2))
                    else
                        (assert (NU 3))
                )
    )
)
```

```
(defrule as-class-1-NK
    (as-class 1)
    (not (NK))=>
    (format t "1 - Достовірний канал повинен використовуватися для початкової ідентифікації і
    автентифікації. Зв'язок з використанням даного каналу повинен ініціюватися виключно
    користувачем%n")
    (format t "2 - Достовірний канал повинен використовуватися для початкової ідентифікації і
    автентифікації та у випадках, коли необхідний прямий зв'язок користувач/КЗЗ або КЗЗ/
    користувач.%n")
    (bind ?response-1 (ask-question "Оберіть одне з двух тверджень: " 1 2))
    (if (eq ?response-1 1)
        then
            (assert (NK 1))
        else
            (assert (NK 2))
    )
)

(defrule as-class-1-NO
    (as-class 1)
    (not (NO))=>

    (if (yes-or-no-p "Політика розподілу обов'язків не визначає дві адміністративні ролі:
    адміністратора безпеки та іншого адміністратора? (так чи ні)")
        then
            (assert (NO 1))
        else

                (if (yes-or-no-p "Політика розподілу обов'язків визначає множину ролей
                користувачів?(так або ні): ")
                        then
                            (assert (NO 3))
                        else
                            (assert (NO 2))
                )
    )
)

(defrule as-class-1-NC
    (as-class 1)
    (not (NC))=>

    (if (yes-or-no-p "Політика цілісності КЗЗ не визначає домен КЗЗ та інші домени, а також
    механізми захисту, що використовуються для реалізації розподілення доменів? (так чи ні)")
        then
            (assert (NC 1))
        else

                (if (yes-or-no-p "КЗЗ гарантує, що послуги безпеки доступні тільки через інтерфейс
                КЗЗ і всі запити на доступ до захищених об'єктів контролюються КЗЗ?(так або ні): ")
                        then
                            (assert (NC 3))
                        else
                            (assert (NC 2))
                )
    )
)
```

```
(defrule as-class-1-NT
    (as-class 1)
    (not (NT))=>
    (format t "1 - Самотестування за запитом%n")
    (format t "2 - Самотестування при старті%n")
    (format t "3 - Самотестування в реальному часі%n")
    (bind ?response-1 (ask-question "Який тип самотестування використовується? (1, 2, 3): " 1 2 3))
    (if (eq ?response-1 1)
        then
            (assert (NT 1))
        else
                (if (eq ?response-1 2)
                    then
                        (assert (NT 2))
                    else
                        (assert (NT 3))
                )
    )
)

(defrule as-class-1-KA
    (as-class 1)
    (not (KA))=>
    (format t "1 - процесу і захищеного об'єкта%n")
    (format t "2 - користувача, процесу і захищеного об'єкта%n")
    (format t "3 - користувача і захищеного об'єкта%n")
    (bind ?response-1 (ask-question "КЗЗ здійснює розмежування доступу на підставі атрибутів доступу
    (1, 2, 3): " 1 2 3))
    (if (eq ?response-1 1)
        then
            (assert (KA 1))
        else
                (if (eq ?response-1 4)
                    then
                        (assert (KA 4))
                    else
                        (if (yes-or-no-p "Політика адміністративної конфіденційності, що
                        реалізується КЗЗ, відносится до всіх об'єктів КС? (так чи ні)")
                    then
                        (assert (KA 3))
                    else
                        (assert (KA 2))
                        )
                )
    )
)

(defrule as-class-1-KO
    (as-class 1)
    (not (KO))=>
    (if (yes-or-no-p "Перш ніж користувач або процес зможе одержати в своє розпорядження звільнений
    іншим користувачем або процесом об'єкт, встановлені для попереднього користувача або процесу
    права доступу до даного об'єкта скасовані? (так чи ні)")
        then
            (assert (KO 1))
    )
)

(defrule as-class-1-CO
    (as-class 1)
    (not (CO))=>
    (format t "1 - певний набір (множину) операцій, виконаних над захищеним об'єктом за певний
    проміжок часу%n")
    (format t "2 - всі операції, виконані над захищеним об'єктом за певний проміжок часу%n")
    (bind ?response-1 (ask-question "Існують автоматизовані засоби, які дозволяють авторизованому
    користувачу або процесу відкатити або відмінити (1, 2)" 1 2))
    (assert (CO ?response-1))
)
```

```
(defrule as-class-1-CA
    (as-class 1)
    (not (CA))=>
    (format t "1 - конкретних користувачів і/або групи користувачів, які мають право модифікувати
    об'єкт%n")
    (format t "2 - конкретні процеси і/або групи процесів, які мають право модифікувати об'єкт%n")
    (format t "3 - конкретні процеси (і групи процесів), які мають, а також тих, які не мають права
    модифікувати об'єкт%n")
    (format t "4 - конкретних користувачів і процеси (і групи користувачів і процесів), які мають, а
    також тих, які не мають права модифікувати об'єкт%n")
    (bind ?response-1 (ask-question "КЗЗ повинен надавати можливість адміністратору або користувачу,
    який має відповідні повноваження, для кожного захищеного об'єкта шляхом керування належністю
    користувачів, процесів і об'єктів до відповідних доменів визначити (1, 2, 3, 4): " 1 2 3 4))
    (assert (CA ?response-1))
)

(defrule as-class-1-DR
    (as-class 1)
    (not (DR))=>
    (if (yes-or-no-p "Політика використання ресурсів, що реалізується КЗЗ, не відноситься до всіх
    об'єктів КС? (так чи ні)")
        then
            (assert (DR 1))
        else
            (bind ?response-2 (ask-question "Політика використання ресурсів повинна визначати
            обмеження, які можна накладати, на кількість даних об'єктів (обсяг ресурсів), що
            виділяються (1 - окремому користувачу, 2 - окремому користувачу або довільним групам
            користувачів)" 1 2))
            (if (eq ?response-2 1)
                then
                    (assert (DR 2))
                else
                    (assert (DR 3))
            )
    )
)

(defrule as-class-1-DV
    (as-class 1)
    (not (DV))=>
    (format t "1 - Після відмови КС або переривання обслуговування КЗЗ повинен перевести КС до
    стану, із якого повернути її до нормального функціонування може тільки адміністратор або
    користувачі, яким надані відповідні повноваження%n")
    (format t "2 - Після відмови КС або переривання обслуговування КЗЗ має бути здатним визначити,
    чи можуть бути використані автоматизовані процедури для повернення КС до нормального
    функціонування безпечним чином.%n")
    (format t "3 - Після будь-якої відмови КС або переривання обслуговування, що не призводить до
    необхідності заново інсталювати КС, КЗЗ повинен бути здатним виконати необхідні процедури і
    безпечним чином повернути КС до нормального функціонування або, в гіршому випадку,
    функціонування в режимі з погіршеними характеристиками обслуговування%n")
    (bind ?response-1 (ask-question "Оберіть правильне твердження (1, 2, 3): " 1 2 3))
    (assert (DV ?response-1))
)
```

```
(defrule as-class-1-DS
    (as-class 1)
    (not (DS))=>
    (if (yes-or-no-p "Політика стійкості до відмов, що реалізується КЗЗ, відноситися до всіх
    компонентів КС? (так чи ні)")
        then
            (if (yes-or-no-p "Відмова одного захищеного компонента призводить до недоступності всіх
            послуг або до зниження характеристик обслуговування? (так чи ні)")
                then
                    (assert (DS 2))
                else
                    (assert (DS 3))
            )
        else
            (assert (DS 1))
    )
)

(defrule as-class-1-DZ
    (as-class 1)
    (not (DZ))=>
    (format t "1 - Політика гарячої заміни, що реалізується КЗЗ, повинна визначати політику
    проведення модернізації КС%n")
    (format t "2 - Політика гарячої заміни, що реалізується КЗЗ, повинна визначати множину
    компонентів КС, які можуть бути замінені без переривання обслуговування%n")
    (format t "3 - Політика гарячої заміни, що реалізується КЗЗ, повинна забезпечувати можливість
    заміни будь-якого компонента без переривання обслуговування%n")
    (bind ?response-1 (ask-question "Оберіть правильне твердження (1, 2, 3): " 1 2 3))
    (assert (DZ ?response-1))
)

(defrule as-1-profile
    (NR 1)
    (NU 1)
    (NK 1)
    (NO 1)
    (NC 1)
    (NT 1)
    =>
    (assert (as-profile 1k1))
    (retract 1)
)
```