УДК 004.9; 004.056/057

[1]**В.В. Казимир**, д.т.н., проф.
[2]**Г.А. Сіра**, асп.
[3]**Д.О. Черних**, магістр

ТЕХНОЛОГІЯ ПОБУДОВИ СИНТЕТИЧНОГО ОТОЧЕННЯ СИСТЕМ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

[1,2]Інститут проблем математичних машин та систем НАН України
[1] E-mail: vvkazymyr@gmail.com
[2] E-mail: seraya.anna@gmail.com
[3]Національний авіаційний університет
[3] E-mail: dim_0n@ukr.net

Запропоновано технологію до формалізованого подання імітаційних моделей систем інформаційної безпеки на основі конвертації формалізованих описів моделей, отриманих за допомогою інших програм моделювання бізнес-процесів синтетичного оточення систем в XML-файли.

архітектура HLA, бізнес-процес, імітаційна модель, синтетичне оточення, система інформаційної безпеки

Вступ

Інтенсивний розвиток інформаційних технологій позитивно вплинув на широкі сфери діяльності суспільства. Однак поруч з цим виникло ряд проблем, пов'язаних з інформаційною безпекою.

Головною метою системи інформаційної безпеки (СІБ) є забезпечення стійкого до зовнішніх впливів функціонування об'єкта, запобігання загрозам його безпеки, створення надійних засобів для надання конфіденційності інформації, зберігання її цілісності та доступності.

За допомогою методів та засобів моделювання можна зробити висновок про стійкість СІБ до зовнішніх впливів.

Процес розроблення СІБ потребує застосування напівнатурного моделювання, яке найкращим чином дозволяє поєднати реально працюючі модулі систем із синтетичним оточенням – програмними імітаторами, які моделюють зовнішні фактори. Синтетичне оточення здатне відтворити найбільш критичні режими роботи СІБ, які у звичайних умовах не проявляються, але мають бути враховані під час їх розроблення.

Аналіз досліджень і публікацій

Найбільш загальні підходи до побудови синтетичного оточення закладено в архітектурі високого рівня High Level Architecture (HLA), розробленій в США наприкінці 90-х років [1].

Архітектура HLA являє собою сукупність методик і стандартів для побудови систем розподіленого моделювання. Ця технологія поєднує системи, побудовані для різних цілей і в різні періоди часу, продукти й платформи різних фірм, дозволяє їм взаємодіяти в єдиному синтетичному оточенні [2]. Для поєднання моделей в HLA використовують інфраструктуру реального часу Run-Time Infrastructure (RTI), яка грунтується на стандарті XML [3]. Разом з тим HLA залишає відкритими питання детального спрацювання формальних методів і практичних рішень, застосовуваних для побудови та реалізації розподілених моделей [4].

Створенню імітаційної моделі передує опис бізнес-процесів [5]. Для цього, зазвичай, застосовують спеціально розроблені програмні засоби, найбільш поширеним з яких є BPwin [6]. Завдяки простій графічній нотації, що використовується в BPwin, до створення моделей бізнес-процесів можуть бути залучені бізнес-аналітики, експерти, системні інтегратори. Це покращує загальне концептуальне висвітлення систем, що моделюються.

Але моделі BPwin є статичні і не можуть відобразити динаміку процесів, тим більше в реальному часі. До того ж вони не забезпечують формального визначення моделей, що робить їх неприйнятними для побудови синтетичного оточення.

Постановка завдання

Актуальним питанням є створення імітаційної моделі СІБ на основі формалізованого опису, який можна отримати за допомогою систем моделювання бізнес-процесів.

Важливими критеріями створення імітаційних моделей є затрачений час та кількість допущених помилок. Використання технології, запропонованої в цій роботі, для формалізованого подання імітаційних моделей на основі конвертації формалізованих описів моделей, отриманих за допомогою інших програм моделювання бізнес-процесів і потоків даних, дозволить значно скоротити час та зменшити потенційну кількість помилок. Формалізацію описів моделей бізнес-процесів проведемо на базі мови XML, що надасть переваги в легкій зміні поданих даних, наочності подання, комутативності, а також створить умови для використання цієї технології на засадах архітектури HLA.

Система імітаційного моделювання JESS

Для створення імітаційних моделей оточення систем інформаційної безпеки пропонується використовувати систему моделювання JESS (Java E-net Simulation System) [7].

У JESS застосовано об'єктно-орієнтовану формальну модель, яка є об'єднанням формальної теорії Е-мереж і теорії агрегатів. Причому агрегати використовуються в цьому випадку як базова концепція структуризації − вони відповідають структурним елементам системи. Е-мережі є найбільш потужним розширенням мереж Петрі, оскільки забезпечують не тільки якісний, але і кількісний аналіз систем, що моделюються. Крім того, формальна теорія Е-мереж дозволяє досить просто здійснювати верифікацію отриманих моделей [8].

Одним з основних елементів JESS є графічна мова специфікацій, що підтримує три рівня моделювання (системи загалом, окремих агрегатів та безпосередньо Е-мережевих переходів).

Технологія конвертації формалізованих описів моделей

Технологічну схему конвертації формалізованих описів моделей у формалізовану імітаційну модель показано на рис. 1.

Основу цієї схеми становить опис моделі мовою XML, виконаний у форматі Petri Net Markup Language (PNML). Цей формат, що відповідає міжнародному стандарту високорівневих мереж Петрі – High-Level Petri Nets (HLPN), у процесі розроблення цієї технології був розширений визначенням Е-мереж і отримав назву CEN PNML. На основі визначення CEN PNML генерується XSD-схема опису моделей мовою XML, яку надалі використовують під час створення і розбору XML-файлів моделей.

Весь процес використання отриманих моделей можна розбити на два етапи. Перший етап − розроблення моделі за допомогою системи моделювання JESS. За спроектованою в графічному редакторі мережі система моделювання автоматично генерує Java-код, який після обробки програмою розбору перетворюється на об'єктну модель (Data Object Model – DOM). Далі викликається бібліотечна функція, яка, використовуючи заздалегідь сформоване XSD-визначення, перетворює об'єктну модель на XML-файл і зберігає даний файл в базі даних (БД).

На другому етапі в разі використання моделі безпосередньо в процесі напівнатурного моделювання потрібний XML-файл береться з БД, зворотним порядком перетворюється в об'єктну модель, а потім і в Java-код, який виконується інтерпретатором моделі.

Функції конвертора тут виконує модуль, який бере на себе завдання сформувати за описом XSD-схеми у форматі CEN PNML файл XML-документа, що містить формалізований опис моделі, а також у разі потреби завантажити цей файл із БД і згенерувати імітаційну модель JESS.

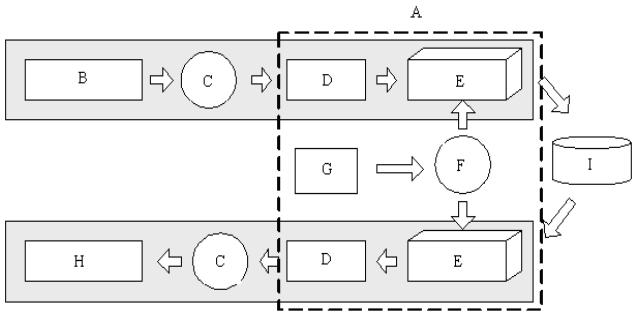

Рис. 1. Технологічна схема конвертації:

A – конвертор;
B – редактор моделі;
C – Java-код;
D – парсер;
E – бібліотека DOM;
F – XSD-схема;
G – визначення CEN PNML;
H – інтерпретатор моделі;
I – XML- уявлення

Конвертація зовнішніх даних

Наведена вище технологія може бути застосована і до інших моделей, які мають подання у вигляді XML. Вона дозволить значно скоротити процес обробки файлу, що імпортується, і максимально використати вже існуючі загальні процедури конвертора, а також знизити затрати на опис процедур для нових форматів файлів, що завантажуються. Для цього немає потреби реалізовувати кожного разу повну процедуру конвертації файлу, що імпортується, до вигляду Java-коду, що сприймається інтерпретатором моделей системи JESS. Необхідно лише звести файл-джерело формату XML-документа до формату даних CEN PNML.

Розглянемо, як працює ця схема у разі використання моделі, сформованої в системі моделювання бізнес-процесів BPWin, яка має змогу експортувати модель бізнес-процесу, побудованого за допомогою діаграм стандарту IDEF3, у XML-файл.

Для вирішення питання про завантаження моделі бізнес-процесу скористаємося вже побудованою схемою (рис. 1) з єдиною відмінністю: у модуль конвертора будемо надсилати замість Java-коду зовнішній файл у форматі XML-схеми IDEF3, а оброблятимемо його згідно з раніше описаним правилам. Конвертований документ поміщується в БД. У разі потреби документ прочитується з БД, обробляється конвертором і далі передається в JESS, зокрема модулю інтерпретатора моделі (рис. 2).

Реалізація необхідної функціональності конвертора передбачає виконання таких етапів:

− робота з даними, що імпортуються;

− збереження даних у спеціалізованому форматі XML;

− конвертація даних із спеціалізованого формату XML;

− відновлення формалізованого опису моделі.

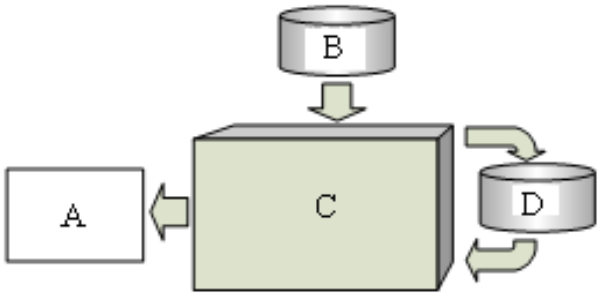

Рис. 2. Схема обробки конвертором зовнішнього файлу:
A – система моделювання JESS;
B – XML-поданнямоделі бізнес-процесу у стандарті IDEF3;
C – конвертор XML-форматів;
D – XML-стандарт

Алгоритм виконання та результати роботи модуля конвертації XML-файлів

Старт процесу конвертації починається із викликом модуля конвертації та подання XML-файлу для оброблення. Далі конвертор має відкрити цей файл і вирішити, яким шляхом треба його перетворювати. Відповідно до варіанта файлу моделі JESS будуть використовуватися методи конвертації JESS та відповідна бібліотека з функцією конвертації. Те саме стосується варіанта конвертації XML-файлу, отриманого з BPWin. Після розбору файлу та виявлення об'єктів моделі будується PNML файл, який зберігається у БД.

Для відображення моделі на екрані файл CEN PNML знов конвертується у файл Java. Отриманий Java-код використовується в JESS для побудови графічної моделі та її виконання інтерпретатором моделі.

Під час конвертації формалізованого опису моделі бізнес-процесу в форматі XML відбувається пошук та ідентифікація необхідних даних про об'єкти моделі на основі закладеної в конверторі інформації про місцеперебування відповідних метаданих.

Після того як усі отримані дані структуровано таким чином, що встановлені позиції, переходи та зв'язки між ними, виконується запис в файл CEN PNML. Елементи Е-мережі записуються у такому порядку: позиції, переходи, дуги. Це відбувається за встановленими правилами в результаті аналізу структури файлів. Всі відповідні властивості заповнюються на основі встановлених відповідностей об'єктів.

Таким чином, алгоритм конвертації дає змогу створити CEN PNML файл за XSD схемою, на яку є посилання в «шапці» створеного файлу «E:\CEN_PNML.xsd».

Наступним етапом в роботі конвертора є етап, що визначений розробленою схемою роботи конвертора як етап відновлення моделі. Він є частиною процесу конвертації як у випадку імпортування, так і у випадку звичайної роботи з імітаційною моделлю JESS.

XML-конвертор отримує файл із БД у форматі CEN PNML і виконує читання об'єктів за правилами:

− спочатку виконується читання всіх позицій моделі, потім всіх визначених переходів і далі всіх дугих;

− створюється Java-код, у якому послідовно записуються стандартні процедури створення моделі у вигляді агрегатів;

− класи агрегатів доповнюються методами додавання відповідних позицій, переходів із визначенням позицій на входах і виходах кожного переходу, а також позицій входу і виходу агрегатів;

− в описі на рівні моделі встановлюється відповідність зв'язків між агрегатами.

Така схема забезпечує значне скорочення додаткових дій у разі потреби додати можливість імпорту нового формату моделі в систему JESS. Адже в цьому разі необхідно лише додати нову процедуру розбирання потрібного файлу моделі та використати відповідну бібліотечну функцію для побудови об'єктної моделі.

Тож XML-конвертор якнайкраще відповідає потребам системи у зв'язку з іншими програмами моделювання процесів. Схема взаємодії JESS із зовнішнім середовищем на базі конвертора є діючою і може застосовуватися для подальшого розроблення та розширення можливостей імпорту даних.

Розроблений алгоритм реалізації відповідає вимогам функціональності та задовольняє потреби системи імітаційного моделювання у коректності та повноті подання даних.

Розглянемо використання запропонованої технології конвертації формальних описів моделей для побудови синтетичного оточення СІБ у вигляді імітаційних моделей, поданих у вигляді Е-мереж.

Синтетичне оточення для систем інформаційної безпеки

Об'єктами оточення СІБ є об'єкти зовнішнього інформаційного простору, що можуть виконувати за допомогою СІБ як пасивний вплив (втрата контакту, зрив зворотного зв'язку тощо), так і активний (різноманітні інформаційні атаки, підміна інформації, додавання непотрібної інформації, зломи паролів тощо). Для прикладу створимо модель інформаційної атаки типу DDoS, оскільки цей клас атак є найбільш критичним за наслідками [9].

DDoS (Distributed Denial of Service) – розподілена відмова в обслуговуванні. Метою атаки є порушення доступності інформаційних ресурсів за допомогою програмних засобів, розміщених на вже вдало атакованих (скомпрометованих) вузлах Internet.

Суть атаки полягає в тому, що одночасно зі всіх скомпрометованих вузлів на об'єкт атаки відправляється велика кількість помилкових запитів і, як наслідок, паралізується робота об'єкта.

Під час проведення DDoS створюється ієрархічна структура агентів атаки – кластер DDoS.

Створену за допомогою системи BPWin у форматі IDEF3 модель бізнес-процесу атаки DDoS показано на рис. 3.

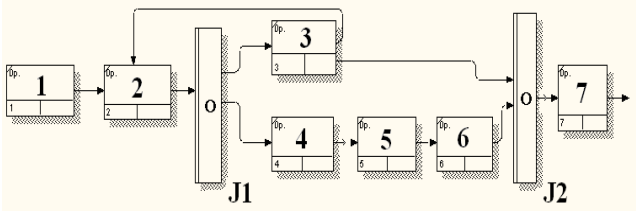

Рис. 3. Модель роботи оточення СІБ у вигляді бізнес-процесу атаки DDoS

Елементи моделі мають таке призначення:

1 – вибрати об'єкт для проведення атаки;

2 – розмістити програмні засоби на вузлах Internet агентів типу 1 (об'єкт ієрархії DDoS, що координує роботу об'єктів нижчого рівня) та агентів типу 2 (об'єкт ієрархії DDoS, що безпосередньо виконує атаку);

3 – відправити повідомлення агентам типу 1 про стан готовності агентів типу 2 до атаки;

4 – зберегти інформацію про стан готовності агентів типу 2;

5 – розподілити навантаження між агентами типу 2;

6 – задати режим атаки (інтенсивність, спосіб підміни IP-адреси);

7 – виконати атаку на об'єкт.

Формалізоване подання імітаційної моделі у вигляді Е-мережі, яке було отримано за допомогою системи імітаційного моделювання JESS на основі запропонованої технології та розробленого модуля конвертації XML, показано на рис. 4. Ця модель може бути використана для тестування процесу функціонування СІБ під впливом інформаційної атаки типу DDoS.

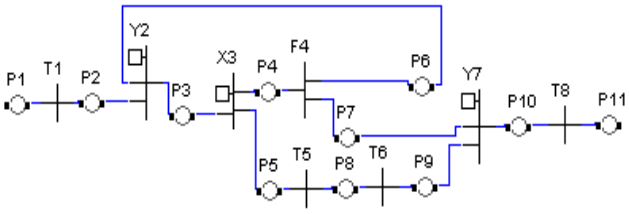

Рис. 4. Формалізоване подання імітаційної моделі у вигляді Е-мережі

Висновки

Запропоновано інформаційну технологію, що забезпечує автоматизоване створення імітаційних моделей шляхом перетворення XML-подань графічних моделей бізнес-процесів, отриманих за допомогою BPwin.

Розроблена технологічна схема та реалізований конвертор XML-файлів дозволяють значно скоротити час та зменшити потенційну кількість помилок під час створення формалізованого подання імітаційних моделей.

Застосування отриманої формалізованої моделі бізнес-процесу у вигляді Е-мережі значно полегшує його тестування шляхом проведення імітаційних експериментів в системі імітаційного моделювання JESS. Після тестування отримані імітаційні моделі можуть бути використані для побудови синтетичного оточення СІБ із метою уточнення їх властивостей та характеристик за допомогою методу напівнатурного моделювання.

Розроблена інформаційна технологія може бути основою для побудови систем розподіленого моделювання в межах реалізації HLA.

Література

1. *Kuhl F.* Creating Computer Simulation Systems: An Introduction to the High Level Architecture / F. Kuhl, R. Weatherly, J. Dahmann. – Prentice Hall PTR, 1999. – 212 p.

2. *Григорьев Р.Н.* Построение распределенных обучающих систем на основе подходов HLA [Електронний ресурс] / Р.Н. Григорьев, П.Б. Панфилов. – Режим доступу до статті: http://miem.net.ru/nit8/e/95.doc.

3. *Самолов Д.В.* Практический опыт построения систем распределенного моделирования на основе архитектуры HLA / Д.В. Самолов, Р.Н. Григорьев, Т.Е. Бах-тина // Авиакосмическое приборостроение. – 2003. – № 9. – С. 51−54.

4. *Казимир В.В.* Моделирование синтетического окружения для реактивных систем / В.В. Казимир // Математичне моделювання. – 2003. – № 2(10). – С. 24–32.

5. *Ситник В.Ф.* Імітаційне моделювання: навчально-методичний посібник для самостійного вивчення дисципліни / В.Ф. Ситник, Н.С. Орленко. – К.: КНЕУ, 1999. – 208 с.

6. *Маклаков С.В.* BPwin и Erwin. CASE-средства разработки информационных систем / С.В. Маклаков . – М.: ДИАЛОГ-МИФИ, 2001. – 304 с.

7. *Kazymyr V.* Application of Java-Technologies for Simulation in the Web / V. Kazymyr, N. Demshevska // Lecture Notes in Informatics (LNI) Proceedings. Series of the German Informatics Society (GI): Bohn. – 2001. – Vol. P-2. – P. 173–184.

8. *Казимир В.В.* Верификация реактивных систем с помощью формул темпоральной логики на Е-сетевых моделях / В.В Казимир // Математичні машини і системи. – 2002. – №1.–С. 29–40.

9. *Сабанов С.Г.* Анатомия хакерской DDoS-атаки [Електронний ресурс] / С.Г. Сабанов. – Режим доступу до статті: http://www.klubok.net/reviews93.html.

Стаття надійшла до редакції 12.11.09.

[1]**В.В. Казимир,** [2]**А.А. Серая,** [3]**Д.А. Черных**

ТЕХНОЛОГИЯ ПОСТРОЕНИЯ СИНТЕТИЧЕСКОГО ОКРУЖЕНИЯ СИСТЕМ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

[1,2]Института проблем математических машин и систем НАН Украины
[3]Национальный авиационный университет

архитектура HLA, бизнес-процесс, имитационная модель, синтетическое окружение, система информационной безопасности

Рассмотрена технология формализированного представления имитационных моделей систем информационной безопасности на основе конвертации формализированных описаний моделей, полученных с помощью других программ моделирования бизнес-процессов синтетического окружения систем в XML-файлы. Имитационные модели могут использоваться для построения синтетического окружения систем информационной безопасности для уточнения их свойств и характеристик. Разработанная информационная технология может служить основой для построения распределенного моделирования в рамках реализации HLA.

[1]**Volodymyr V. Kazymyr,** [2]**Hanna A. Sira,** [3]**Dmitro O. Chernyh**

THE TECHNOLOGY OF SYNTHETIC ENVIRONMENT CONSTRUCTING FOR INFORMATION SECURITY SYSTEMS

[1,2]Institute of the Problems of the Mathematical Machines and Systems of the National Academy of Sciences Ukraine
[3]National Aviation University

business process, HLA architecture, information security system, simulated model, synthetic environment

An information technology is suggested to provide the automatic development of simulated models by converting graphic models of business process, got by other programs for modeling. Got simulated models can be used for the synthetic environment construction of information security systems to define more exactly models features and characteristics. Developed information technology can be as base for distributed modeling systems constructing within the limits of realization HLA.