Г.М. Саксонов, О.А. Жукова, Ю.А. Мілінчук

Національний технічний університет “Дніпровська політехніка”, Дніпро

СИСТЕМА ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ КРИПТОГРАФІЧНИХ ПРОТОКОЛІВ

У статті розглянута клієнт-серверна система для імітаційного моделювання роботи криптографічних протоколів і можливих атак на них. Запропоновано структуру системи моделювання, принцип формування компонент криптографічного протоколу і порядок роліової взаємодії його учасників. Представлена програмна реалізація редактора протоколів, який дозволяє сформувати символічну, вербальну і обчислювальну специфікації. Запропоновано рекомендації щодо застосування системи.

Ключові слова: клієнт-серверна система, імітаційне моделювання, криптографічний протокол, криптографічний алгоритм, програмна реалізація, редактор протоколу.

Вступ

Постановка проблеми. Процес розробки протоколів криптографічних систем складається з перевірки його стійкості до атак в припущенні про надійність криптографічних примітивів, на які він спирається [1]. Найчастіше криптосистеми, що використовуються, не є надійними і практично всі в тій чи іншій мірі уразливі [2–3]. При цьому передбачається, що ненадійність криптографічних систем викликана або наявністю в них помилок при реалізації криптоалгоритмів, або їх неправильним застосуванням. Теза про потенційну ненадійність криптосистем підкріплюється і відсутністю повних формальних критеріїв якості криптосистем, і неможливістю теоретичного висновку про достатній рівень захисту [4–8]. Сервісні програми, що використовуються для тестування, регулярно виявляють проблеми в існуючих протоколах. Однак з їх допомогою не можна «довести» надійність криптографічної системи [11–12]. Як правило, надійність більшості криптографічних систем заснована на результатах багаторічного практичного криптоаналізу системи, в тій чи іншій мірі підкріплених математичним обґрунтуванням [3]. Практичний аналіз доповнює формальний і відбувається із залученням експертів, які здійснюють роліове виконання протоколу. По суті, такий аналіз – це тестування обраного протоколу в середовищі інформаційного обміну [7–8]. Створенню такого дослідницького середовища присвячена ця стаття.

Аналіз останніх досліджень і публікацій. У роботах [13–14] представлені можливості практичного аналізу протоколів з використанням візуальних середовищ моделювання. Візуальна модель сприяє

цілісному сприйняттю протоколів що модулюються, дозволяє уявити протокол у формі діаграм, що показують його учасників і їх можливості (статичне представлення), а також процеси взаємодії між учасниками протоколу (динамічне представлення). Протокол декомпонується на ряд обчислювальних алгоритмів і послідовність повідомлень, а учасники протоколу об'єднуються в сеанс, в рамках якого кожен з них виконує свою роль. При цьому, можливість розподіленого виконання обчислювальних операцій учасниками протоколу є важливою додатковою перевагою такого середовища моделювання.

Метою статті є опис розробленої клієнт-серверної системи імітаційного моделювання криптографічних протоколів, що дозволяє уявити у спрощеній інтуїтивно зрозумілій формі роліову взаємодію учасників цього протоколу. Розроблена система являє собою протокол у формі візуальної діаграми обміну повідомленнями між учасниками.

Виклад основного матеріалу

Призначення системи. Запропонована система призначена для моделювання так званих навчальних протоколів, які демонструють тільки принцип побудови роботи протоколів в реальних захищених криптографічних системах інформаційного обміну. Це обумовлено тим, що в таких системах, крім криптографічних методів, що забезпечують різні види аутентифікації і безпеки переданих повідомлень, використовуються прийоми, спрямовані проти неправильної їх реалізації або впливу людського фактору (безпеку кодування повідомлень, виявлення і підтвердження помилок і т.п.). Тому, крім навчальної мети, реалізація власних протоколів і моделювання їх роботи в розподіленій інфраструктурі про-

понованій системи моделювання, демонструє всю складність проблеми побудови надійних криптосистем.

Структура системи. Процес розробки протоколу в запропонованій системі складається з двох послідовних етапів – опису протоколу і моделювання його роботи учасниками протоколу. Перший етап розробки проводиться за допомогою редактора протоколу, другий – з використанням середовища моделювання роботи протоколу.

Редактор протоколу – додаток, що дозволяє розробити специфікації описуваного криптографічного протоколу. Ці специфікації містять вербальний та символічний опис всіх перетворень, які виконуються учасниками протоколу, що дозволяє сформувати візуальну діаграму обміну повідомлень між учасниками. Користувач цієї програми, використовуючи спеціальну нотацію, виконує всі необхідні перетворення для двох учасників протоколу, описує обмін повідомленнями між ними, тим самим вирішуючи завдання першого етапу – опис криптографічного протоколу. Виконання кожної операції супроводжується її чисельною реалізацією, яка підтверджує правильність виконання. Отримані таким чином специфікації є інструкцією для всіх користувачів другої програми – середовища моделювання роботи протоколу.

Середовище моделювання являє собою розподілений клієнт-серверний додаток, два користувача якого, незалежно один від одного, виконують функції учасників протоколу за розробленою інструкцією. Третій незалежний користувач, використовуючи специфікації протоколу, реалізує функції противника (порушника). Противник протоколу – суб'єкт, який спостерігає за переданими повідомленнями, і, можливо, втручається в роботу учасників шляхом перехоплення, спотворення (модифікації), вставки (створення нових), повтору і переспрямування повідомлень, блокування передачі і т.п. з метою порушення однієї або декількох функцій безпеки протоколу [2]. Таким чином, спільна незалежна робота всіх користувачів цього додатка призводить до імітації роботи протоколу в умовах наявності активного противника. Описані специфікації повинні бути доповнені назвою протоколу і переліком його компонент, отримання яких порушником вважається фактом злому протоколу.

На рис. 1 умовно показані користувачі системи (Аліса і Боб), які виконують ролі учасників протоколу і користувач, виконуючий роль порушника протоколу (Меллорі).

Основою всієї системи розробки є модуль виконання операцій (рис. 1). Цей модуль використовується всіма додатками системи і призначений для реалізації криптографічних алгоритмів, обчислювальних та інших операцій перетворення даних прото-

колу. При цьому передбачається абсолютна надійність криптографічних алгоритмів і методів, що використовуються в цьому модулі. Цей модуль має уніфікований інтерфейс, що дозволяє використовувати його у всіх додатках системи. Послідовне застосування необхідних операцій цього модуля дозволяє користувачеві редактора сформувати всі специфікації протоколу, а користувачам середовища моделювання – необхідні повідомлення.

Рис. 1. Структура системи моделювання

Користувачі клієнтських додатків намагаються виконати свою частину протоколу, використовуючи його відому специфікацію, а користувач серверного додатка, якому також відома ця специфікація, виконує емуляцію атаки на цей протокол. Для користувача цієї програми повинна бути вказана мета моделювання – порушення секретності або перевірка справжності. Секретність пов'язана з певними даними користувача, а перевірка справжності (аутентифікація) пов'язана з двома користувачами і їх даними. Користувач серверного додатка відстежує вхідні повідомлення від клієнтських додатків і при отриманні повідомлення від одного з них може виконати одну з наступних дій:

- передати отримане повідомлення без зміни;
- довільно або за допомогою операцій модуля змінити отримане повідомлення і передати його іншому клієнтському додатку;
- ігнорувати це повідомлення і не передавати його.

Використання таких команд дозволяє реалізувати будь-які дії порушника протоколу: прослуховування, заміну повідомлення, переривання передачі

або виконання передачі повідомлення без втручання. Крім цього у користувача, який виконує роль порушника, є можливість використовувати будь-які перехоплені дані поточного або попередніх сеансів протоколу. При моделюванні роботи протоколу можуть бути виконані кілька сесій. На початку кожної сесії компоненти протоколу, сформовані клієнтськими додатками (учасниками протоколу), зникають, а дані, отримані порушником протоколу, накопичуються і можуть бути використані в наступних сесіях.

Програмна реалізація. Розроблено пілотну версію редактора системи, в якому побудова протоколу проводиться у разі такої концепції формування його компонентів:

- протокол відтворюється послідовністю обчислювальних операцій і операціями передачі їх результатів учасникам протоколу;
  - кожна обчислювальна операція характеризується своїм уніфікованим ідентифікатором, що визначає тип і значення даних, отриманих в результаті програмної реалізації цієї операції. Фактично цей ідентифікатор операції є унікальною змінною програми, що зберігає певне значення;
  - обмін повідомлень в протоколі фактично означає передачу відповідного ідентифікатора іншому учаснику протоколу;
  - кожен учасник протоколу має свій набір ідентифікаторів операцій, що складаються з власних ідентифікаторів або отриманих від іншого учасника;
  - ідентифікатор операції може мати параметри - будь-які інші ідентифікатори, які учасники протоколу отримали в результаті виконання попередніх операцій.
- Ідентифікатор обчислювальної операції має спеціальну нотацію, яка визначає:
- номер кроку протоколу, на якому ця операція була виконана;
  - скорочене позначення імені учасника протоколу (наприклад - А: Аліса);
  - ім'я типу даних, одержаних в результаті виконання обчислювальної операції;
  - параметри (вихідні дані для виконання операції).

Наприклад, ідентифікатор `Enc.4_A (Mes.2_B)` означає, що він отриманий на 4-му кроці протоколу в результаті виконання Алісою операції асиметричного шифрування (`Enc`) даних ідентифікатора `Mes.2_B`. Для більш інформативного сприйняття деякі ідентифікатори позначені відповідними графічними іконками. При цьому ідентифікатор, отриманий при розшифровці шифротекста, збігається з ідентифікатором, який шифрується.

У даній версії програми учасникам системи доступні процедури формування текстових повідомлень, передачі даних іншому учаснику протоколу, упаковки різномірних даних в одне повідомлення і

функція порівняння даних (рис. 2). Кожен учасник протоколу має в своєму розпорядженні набір криптографічних алгоритмів (асиметричного і симетричного шифрування), процедури генерування відповідних ключів шифрування і функцію хешування. При цьому в одному сеансі учаснику протоколу може бути надана тільки одна пара симетричних ключів і один сеансовий ключ асиметричного шифрування.

Рис. 2. Меню вибору операцій учасником протоколу

При виконанні користувачем будь-якої операції з цього списку автоматично виконується наступна послідовність дій:

- символічне відображення поточної операції на діаграмі обміну повідомленнями;
- обчислювальні результати цієї операції і додавання їх в список;
- формування вербального опису операції.

Рис. 3. Інтерфейс діалогового вікна для формування параметрів операцій протоколу

Формування протоколу здійснюється з використанням інтерфейсу Point-and-click («вказав і клац-

нув»), який не дозволяє користувачеві некоректно використовувати ключі для різних методів шифрування. При цьому користувачеві необхідно лише вибрати учасника протоколу, вид операції та необхідні параметри цієї операції (рис. 3).

На рис. 4 показана візуальна діаграма обміну повідомленнями для опису фрагмента протоколу підпису документа за допомогою криптографії відкритих ключів.

Рис. 4. Візуальна діаграма обміну повідомленнями

- (1) Аліса сформувала текстовое сообщение Mes.1_A
Передает Бобу это сообщение
- (2) Аліса вычисляет хеш-код Mes.1_A
- (3-4) Аліса получила асимметричные ключи
Передает Бобу ключ для дешифрования Kd_A
- (5) Аліса шифрует hash.2_A(Mes.1_A) ключом Ke_A
Посылает этот шифротекст Бобу
- (6) Боб расшифровывает шифротекст Enc.5_A(hash.2_A(Mes.1_A)) ключом Kd_A
- (7) Боб вычисляет хеш-код Mes.1_A

Рис. 5. Вербальний опис фрагменту протоколу

Вербальний опис цього фрагмента і результати чисельного виконання кожної операції протоколу наведені на рис. 5 і рис. 6

Рис. 6. Результати чисельного виконання операцій протоколу

Висновки

В роботі розглядається клієнт-серверна система, що дозволяє модулювати динаміку роботи криптографічного протоколу і можливих атак на нього. Використання системи для незалежного виконання користувачами кроків протоколу, дозволяє проводити неформальну оцінку його надійності при різних сценаріях дій злоумисника, сприяють цілісному сприйняттю модельованих протоколів і дозволяє представити протокол у формі діаграм, відобразити учасників протоколу і їх можливості.

Визначена структура системи моделювання, концепція формування компонент криптографічного протоколу і порядок роліової взаємодії його учасників (користувачі і порушник).

Основою всієї системи розробки є модуль виконання операцій.

Послідовне застосування необхідних операцій цього модуля дозволяє користувачеві редактора сформулювати всі специфікації протоколу, а користувачам середовища моделювання – необхідні повідомлення.

Розроблена система моделювання дає можливість її використання для формування практичних навичок при вивченні ідей і методів сучасної теорії криптографічних протоколів.

Тестування візуального редактора, використаного в якості інструментарію, підтверджує правильність запропонованої концепції формування компонент протоколу.

Його застосування дозволяє зрозуміти які структурні елементи криптографії використані в протоколі, яким чином описана логіка їх взаємодії і дає можливість наочно простежити кроки протоколу, що підвищує зручність його сприйняття, розуміння і аналіз.

Список літератури

1. Котенко И.В. Верификация протоколов безопасности на основе комбинированного использования существующих методов и средств / И.В. Котенко, С.А. Резник, А.В. Шоров // Труды Санкт-Петербургского института информатики и автоматизации. – 2009. – № 8. – С. 292-310.
2. Касперски К. Техника и философия хакерских атак – записки мыш'а / К. Касперски. – М.: СОЛОН-Пресс, 2004. – 272 с.
3. Терехов А.Н. Криптография с открытым ключом: от теории к стандарту / А.Н. Терехов // Программирование РАН. – 1994. – № 5. – С. 17-22.
4. Черемушкин А.В. Криптографические протоколы: основные свойства и уязвимости / А.В. Черемушкин // Прикладная дискретная математика. – 2009. – № 2. – С. 115-150.
5. Семьянов П. Почему криптосистемы ненадежны? [Электронный ресурс] / П. Семьянов. – Центр Защиты Информации. – Режим доступа: <http://algolist.ru/defence/attack/whycrypt.php>.
6. Yan J. A Formal Analysis Method with Reasoning for Cryptographic Protocols / J. Yan, K. Wagatsuma, H. Gao, J. Cheng // 12th International Conference on Computational Intelligence and Security (CIS). – Wuxi, 16-19 December 2016. – P. 566-570. – Available at: <https://ieeexplore.ieee.org/document/7820529?denied=>.

7. Фергюсон Н. Практическая криптография / Н. Фергюсон, Б. Шнайер. – М.: Диалектика, 2017. – 420 с.
8. Liu S. Computational Rationality of Formal Analysis on Cryptographic Protocols / S. Liu, J. Ye // Fourth International Conference on Computational and Information Sciences. – Chongqing, 17-19 August 2012. – pp. 904-907. – Available at: <https://ieeexplore.ieee.org/document/6300530?denied=>.
9. Косачев А.С. Анализ подходов к верификации функций безопасности и мобильности / А.С. Косачев, В.Н. Пономаренко. – М.: Институт системного программирования РАН, 2004. – 101 с.
10. Шнайер Б. Слабые места криптографических систем [Электронный ресурс] / Б. Шнайер. – Центр Защиты Информации. – Режим доступа: <http://algolist.ru/defence/attack/cryptholes.php>.
11. Шнайер Б. Секреты и ложь. Безопасность данных в цифровом мире / Б. Шнайер. – СПб.: Питер, 2003. – 368 с.
12. Черемушкин А.В. Автоматизированные средства анализа протоколов / А.В. Черемушкин // Прикладная дискретная математика. – 2009. – № 1. – С. 34-36.
13. Михайлов А.С. Итерационный процесс разработки протоколов информационного обмена / А.С. Михайлов, С.С. Селезнёв // Прикладная информатика. – № 5(23). – 2009. – С. 59-67.
14. Бабенко Л.К. Верификация безопасности протокола электронной цифровой подписи с помощью AVISPA / Л.К. Бабенко, Х.А. Санчес Россель // Вопросы кибербезопасности. – 2017. – № 2(20). – С. 51-52.

References

1. Kotenko, I.V. (2009), "Verifikaciya protokolov bezopasnosti na osnove kombinirovannogo ispol'zovaniya sushchestvuyushchih metodov i sredstv" [Verification of Security Protocols Based on the Combined use of Existing Methods and Tools], *Proceedings of the St. Petersburg Institute of Informatics and Automation*, No. 8, pp. 292-310.
2. Kasperski, K. (2004), "Tekhnika i filozofiya hakerskih atak – zapiski myshch'a" [Technique and Philosophy of Hacker Attacks - Mouse Notes], SOLON-Press, Moscow, 272 p.
3. Terekhov, A.N. (1994), "Kriptografiya s otkrytym klyuchom: ot teorii k standartu" [Public Key Cryptography: From Theory to the Standard], *Programming of the Russian Academy of Sciences*, No. 5, pp. 17-22.
4. Cheremushkin, A.V. (2009), "Kriptograficheskie protokoly: osnovnye svoystva i uyazvimosti" [Cryptographic Protocols: Key Features and Vulnerabilities], *Applied Discrete Mathematics*, No. 2, pp. 115-150.
5. Sem'yanov, P. "Pochemu kryptosistemy nenadezhny?" [Why are Cryptosystems Unreliable?], *Information Protection Center*, available at: <http://algolist.ru/defence/attack/whycrypt.php>
6. Yan, J., Wagatsuma, K., Gao, H. and Cheng, J. (2016), A Formal Analysis Method with Reasoning for Cryptographic Protocols, *12th International Conference on Computational Intelligence and Security (CIS)*, 16-19 December, Wuxi, China, pp. 566-570, available at: <https://ieeexplore.ieee.org/document/7820529?denied=>.
7. Ferguyson, N. and Shnajer, B. (2017), "Prakticheskaya kriptografiya" [PracticalCryptography], *Dialektika*, Moscow, 420 p.
8. Liu, S. and Ye, J. (2012), Computational Rationality of Formal Analysis on Cryptographic Protocols, *Fourth International Conference on Computational and Information Sciences*, 17-19 August, Chongqing, China, pp. 904-907, available at: <https://ieeexplore.ieee.org/document/6300530?denied=>.
9. Kosachev, A.S. and Ponomarenko, V.N. (2004), "Analiz podhodom k verifikacii funkciy bezopasnosti i mobil'nosti" [Analysis of Approaches to Verification of Security and Mobility Functions], *Institut sistemnogo programmirovaniya RAN*, Moscow, 101 p.
10. Shnajer, B. "Slabye mesta kriptograficheskikh system" [Weaknesses of cryptographic systems], *Information Protection Center*, available at: <http://algolist.ru/defence/attack/cryptholes.php>.
11. Shnajer, B. (2003), "Sekrety i lozh. Bezopasnost dannyh v cifrovom mire" [Secrets and lies. Data Security in the Digital World], SPb, Sankt-Peterburg, 368 p.
12. Cheremushkin, A.V. (2009), "Avtomatizirovannye sredstva analiza protokolov" [Automated protocol analysis tools], *Applied Discrete Math*, No. 1, pp. 34-36.
13. Mihajlov, A.C. and Seleznyov, C.C. (2009), "Iteracionnyj process razrabotki protokolov informacionnogo obmena" [Iterative process of developing information exchange protocols], *Applied Informatics*, No. 5(23), pp. 59-67.
14. Babenko, L.K. and Sanches Rossel', H.A. (2017), "Verifikaciya bezopasnosti protokola elektronnoj cifrovoj podpisi s pomoshch'yu AVISPA" [Security Verification of Electronic Digital Signature Protocol Using AVISPA], *Cybersecurity Issues*, No. 2(20), pp. 51-52.

Надійшла до редколегії 31.03.2020

Схвалена до друку 14.04.2020

Відомості про авторів:

Саксонов Геннадій Михайлович
старший викладач
Державного технічного університету
"Дніпровська політехніка",
Дніпро, Україна
<https://orcid.org/0000-0002-9181-1401>

Information about the authors:

Hennadii Saksonov
Senior Instructor
of Dnipro University
of Technology
Dnipro, Ukraine
<https://orcid.org/0000-0002-9181-1401>

Жукова Олена Андріївна

старший викладач Державного технічного університету
“Дніпровська політехніка”,
Дніпро, Україна
<https://orcid.org/0000-0002-7994-4180>

Мілінчук Юлія Анатоліївна,

асистентка Державного технічного університету
“Дніпровська політехніка”,
Дніпро, Україна
<https://orcid.org/0000-0003-1713-661X>

Olena Zhukova

Senior Instructor
of Dnipro University of Technology
Dnipro, Ukraine
<https://orcid.org/0000-0002-7994-4180>

Uliya Milinchuk

Assistant Lecturer
of Dnipro University of Technology
Dnipro, Ukraine
<https://orcid.org/0000-0003-1713-661X>

СИСТЕМА ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ КРИПТОГРАФИЧЕСКИХ ПРОТОКОЛОВ

Г.М. Саксонов, Е.А. Жукова, Ю.А. Милинчук

В статье рассмотрена клиент-серверная система для имитационного моделирования работы криптографических протоколов и возможных атак на них. Предложена структура системы моделирования, принцип формирования компонент криптографического протокола, и порядок ролевого взаимодействия его участников. Представлена программная реализация редактора протоколов, позволяющего сформировать символьную, вербальную и вычислительную спецификации. Предложены рекомендации по применению системы.

Ключевые слова: клиент-серверная система, имитационное моделирование, криптографический протокол, криптографический алгоритм, программная реализация, редактор протокола.

SYSTEM OF SIMULATION MODEL OF CRYPTOGRAPHIC PROTOCOLS

H. Saksonov, O. Zhukova, U. Milinchuk

An overview of a client-server system that allows to simulate the dynamics of the cryptographic protocol and possible attacks on it is given in the work. In this system, the protocol is decomposed into a series of computational algorithms and a sequence of messages, and the protocol participants are connected into a session, so each of them performs its role within this session. Using users of the system for independent performing of protocol steps allows to do an informal assessment of its reliability under various scenarios of an attacker's actions, also contributes to a holistic perception of the modeled protocols and allows the protocol to be presented in the form of diagrams which show the protocol participants and their capabilities. The structure of the modeling system, the concept of how the components of the cryptographic protocol are formed and the order of the role interaction of its participants (users and the intruder) are described. The basis of the entire development system is the operations module. At the same time, the absolute reliability of cryptographic algorithms and methods used in this module are assumed. The module has a unified interface that allows you to use it in all applications of the system. The consistent application of the necessary operations of this module allows the user of the editor to generate all the protocol specifications, and the users of the modeling environment can generate the necessary messages. The user interface of this application and recommendations for its practical application are proposed. The testing of the visual editor confirms that proposed concept of the protocol components formation is correct. Its application allows us to understand what structural elements of cryptography are used in the protocol, how the logic of their interaction is described and makes it possible to visually trace the steps of the protocol, which increases the convenience of its perception, understanding and analysis.

Keywords: client-server system, simulation, cryptographic protocol, cryptographic algorithm, software implementation, protocol editor.