МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
РАДІОЕЛЕКТРОНІКИ

МАТЕРІАЛИ ПЕРШОГО МІЖНАРОДНОГО НАУКОВО-ПРАКТИЧНОГО ФОРУМУ

«GLOBAL CYBER SECURITY FORUM»

14 – 16 листопада 2019 р.

Харків 2019

Перший міжнародний науково-практичний форум «Global Cyber Security Forum». Зб. матеріалів форуму. – Харків: ХНУРЕ. 2019. – 115 с.

В збірник включені матеріали Першого міжнародного науково-практичного форуму «Global Cyber Security Forum».

Видання підготовлено кафедрою безпеки інформаційних технологій
Харківського національного університету радіоелектроніки

61166 Україна, Харків, просп. Науки, 14
тел./факс: (057) 7021397

E-mail: gcsforum@nure.ua

ГОЛОВА ПРОГРАМНОГО КОМІТЕТУ

СЕМЕНЕЦЬ В.В. – д.т.н., проф., ректор Харківського національного університету радіоелектроніки *(м. Харків, Україна)*.

ЗАСТУПНИКИ ГОЛОВИ ПРОГРАМНОГО КОМІТЕТУ

ХАЛІМОВ Г.З. – д.т.н., проф., зав. каф. безпеки інформаційних технологій Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
НЕОФІТНИЙ М.В. – к.т.н., доц., проректор з наукової роботи Харківського національного університету радіоелектроніки *(м. Харків, Україна).*

ПРОГРАМНИЙ КОМІТЕТ

JAMES VANDIVER – Cybersecurity Engineer at NATO *(USA).*
АНТІПОВ І.Є. – д.т.н., проф., зав. каф. комп'ютерної радіоінженерії і систем технічного захисту інформації Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
БАРАБАШ О.В. – д.т.н., проф., зав. каф. вищої математики Державного університету телекомунікацій *(м. Київ, Україна).*
ВАСІЛІУ Є.В. – д.т.н., проф., директор навчально-наукового інституту «Радіо, телебачення та інформаційної безпеки» Одеської національної академії зв'язку ім. О.С. Попова *(м. Одеса, Україна).*
ГАВВА Д.С. – к.т.н., доц. каф. комп'ютерної радіоінженерії і систем технічного захисту інформації Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ГОРБЕНКО І.Д. – д.т.н., проф. каф. безпеки інформаційних систем і технологій Харківського національного університету ім. В.Н. Каразіна *(м. Харків, Україна).*
ГОРЕЛОВ Д.Ю. – к.т.н., доц., заст. декана факультету інформаційних радіотехнологій і технічного захисту інформації Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ГРЕБЕННІК І.В. – д.т.н., проф., зав. каф. системотехніки Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ДУДАР З.В. – к.т.н., проф., зав. каф. програмної інженерії Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ЄРОХІН А.Л. – д.т.н., проф., декан факультету комп'ютерних наук Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ЖУКОВИЦЬКИЙ І.В. – д.т.н., проф., зав. каф. електронних обчислювальних машин Дніпровського національного університету залізничного транспорту ім. академіка В. Лазаряна *(м. Дніпро, Україна).*
ЗАДІРАКА В.К. – д.ф-м.н., проф., академік НАН України, Інститут кібернетики ім. В.М. Глушкова НАН України *(м. Київ, Україна).*
КАРТАШОВ В.М. – д.т.н., проф., зав. каф. медіаінженерії та інформаційних радіоелектронних систем Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
КАТЕРИНЧИК Р.О. – CEO & Founder at ArtjokeR, член наглядової ради Kharkiv IT Cluster *(м. Харків, Україна).*
КОБЗЄВ В.Г. – к.т.н., доц. каф. Прикладної математики Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
КОРЧЕНКО О.Г. – д.т.н., проф., зав. каф. безпеки інформаційних технологій Національного авіаційного університету *(м. Київ, Україна).*

КУЗНЕЦОВ О.О. – д.т.н., проф. каф. безпеки інформаційних систем і технологій Харківського національного університету ім. В.Н. Каразіна *(м. Харків, Україна)*.
ЛЕВИКІН В.М. – д.т.н., проф., зав. каф. інформаційних управляючих систем Харківського національного університету радіоелектроніки *(м. Харків, Україна)*.
ЛІТКЕВИЧ Д.О. – засновник компанії-співорганізатора **GCS FORUM** ID Clobal Consulting Group *(м. Харків, Україна)*.
ЛУЖЕЦЬКИЙ В.А. – д.т.н., проф., зав. каф. захисту інформації Вінницького національного технічного університету *(м. Вінниця, Україна).*
ЛЯШЕНКО О.С. – к.т.н., доц., декан факультету комп'ютерної інженерії та управління Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
МАКСИМОВИЧ В.М. – д.т.н., проф., зав. каф. безпеки інформаційних технологій Національного університету «Львівська політехніка» *(м. Львів, Україна)*.
МАЧУСЬКИЙ Є.А. – д.т.н., проф., зав. каф. фізико-технічних засобів захисту інформації Національного технічного університету «Київський політехнічний інститут» *(м. Київ, Україна).*
МАШТАЛІР В.П. – д.т.н., проф. каф. інформатики Харківського національного університету радіоелектроніки *(м. Харків, Україна)*.
МІХАЛЬ О.П. – д.т.н., доц., зав. каф. електронних обчислювальних машин Харківського національного університету радіоелектроніки *(м. Харків, Україна)*.
ОЛЕЙНІКОВ А.М. – к.т.н., проф. каф. комп'ютерної радіоінженерії і систем технічного захисту інформації Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ОЛІЙНИКОВ Р.В. – д.т.н., доц., проф. каф. безпеки інформаційних систем і технологій Харківського національного університету імені В.Н. Каразіна *(м. Харків, Україна)*.
ПОТІЙ О.В. – д.т.н., проф. каф. безпеки інформаційних систем і технологій Харківського національного університету ім. В.Н. Каразіна *(м. Харків, Україна).*
РУЖЕНЦЕВ В.І. – д.т.н., доц., проф. каф. безпеки інформаційних технологій Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
САВЧЕНКО В.А. – д.т.н., проф., зав. каф. систем інформаційного та кібернетичного захисту Державного університету телекомунікацій *(м. Київ, Україна).*
САКАЛО С.М. – к.т.н., доц., декан факультету інформаційних радіотехнологій і технічного захисту інформації Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
СЕМЕНОВ С.Г. – д.т.н., проф., зав. каф. обчислювальної техніки та програмування Національного технічного університету «Харківський політехнічний інститут» *(м. Харків, Україна).*
СНІГУРОВ А.В. – к.т.н., доц., декан факультету інфокомунікацій Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ТРОФІМЕНКО В.М. – к.юр.н., доц. каф. кримінального процесу Національного юридичного університету ім. Ярослава Мудрого *(м. Харків, Україна).*
ТУРУТА О.П. – к.т.н., доц. каф. програмної інженерії Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ТОЛЮПА С.В. – д.т.н., проф. каф. кібербезпеки та захисту інформації Київського національного університету ім. Т.Г. Шевченка *(м. Київ, Україна)/*
ФІЛАТОВ В.О. – д.т.н., проф., зав. каф. штучного інтелекту Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ХАРЧЕНКО В.С. – д.т.н., проф., зав. каф. комп'ютерних систем, мереж і кібербезпеки Національного аерокосмічного університету ім. М.Є. Жуковського «Харківський авіаційний інститут» *(м. Харків, Україна).*

ЦОПА О.І. – д.т.н., проф., зав. каф. радіотехнологій інформаційно-комунікаційних систем Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ШОНІЯ О.Б. – д.т.н., проф., академік РАЕН, департамент організаційного управління Грузинського технічного університету *(м. Тбілісі, Грузія).*

СПІВГОЛОВИ ОРГАНІЗАЦІЙНОГО КОМІТЕТУ

ХАЛІМОВ Г.З. – д.т.н., проф., зав. каф. безпеки інформаційних технологій Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ЛІТКЕВИЧ Д.О. – засновник компанії-співорганізатора **GCS FORUM** ID Clobal Consulting Group *(м. Харків, Україна)*.

ОРГАНІЗАЦІЙНИЙ КОМІТЕТ

ГОРЕЛОВ Д.Ю. – к.т.н., доц., заст. декана факультету інформаційних радіотехнологій і технічного захисту інформації Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ЄРОХІН А.Л. – д.т.н., проф., декан факультету комп'ютерних наук Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
КОЗІДУБОВА І.В. – директор компанії-співорганізатора **GCS FORUM** ID Clobal Consulting Group *(м. Харків, Україна)*.
ЛЕБІДЬ В.В. – начальник відділу внутрішнього аудиту Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
ЛЯШЕНКО О.С. – к.т.н., доц., декан факультету комп'ютерної інженерії та управління Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
МЕДВЕДЄВ Є.О. – заст. декана факультету інформаційних радіотехнологій і технічного захисту інформації Харківського національного університету радіоелектроніки *(м. Харків, Україна).*
САКАЛО С.М. – к.т.н., доц., декан факультету інформаційних радіотехнологій і технічного захисту інформації Харківського національного університету радіоелектроніки *(м. Харків, Україна).*

ВЧЕНИЙ СЕКРЕТАР ФОРУМУ

МЕДВЕДЄВ Є.О. – заст. декана факультету інформаційних радіотехнологій і технічного захисту інформації Харківського національного університету радіоелектроніки *(м. Харків, Україна).*

Проблема стандартизації криптографічних перетворень в перехідний та постквантовий періоди та стан її вирішення

І. Д. Горбенко[1,2], О. Г. Качко[2],
Ю. І. Горбенко[2], М. В. Єсіна[1,2],
В. А. Пономар[2]

1. Кафедра безпеки інформаційних систем і технологій, Харківський національний університет імені В.Н. Каразіна, Україна, Харків, пл. Свободи, 6, E-mail: gorbenkoi@iit.kharkov.ua, m.v.yesina@karazin.ua

2. АТ «Інститут інформаційних технологій», Україна, Харків, вул. Бакуліна, 12, E-mail: iit@iit.kharkov.ua, gorbenkou@iit.kharkov.ua, Laedaa@gmail.com

Коротка анотація – Розглядаються та аналізуються загрози, що мають бути реалізовані у постквантовий період стосовно існуючих стандартів криптографічних перетворень. Обґрунтовуються вимоги до них. Формулюється проблема розробки, стандартизації та застосування постквантових стандартів, наводиться аналіз стану та визначаються шляхи її вирішення.

Ключові слова – вимоги до постквантових перетворень, криптографічна стійкість та складність, перспективні методи асиметричних перетворень, стан розробки стандартів та їх порівняння, шляхи впровадження.

I. Вступ

Нині, та очевидно, в деякій перспективі, для криптографічного захисту інформації застосовуються та будуть застосовуватись методи та алгоритми криптоперетворень, що орієнтовані на застосування існуючих, стандартизованих. Вони є суттєвою складовою забезпечення кібербезпеки. Але є обґрунтовані підозри, що у постквантовий період існуючі стандарти асиметричних криптоперетворень, будуть зламуватись за допомогою квантових криптоаналітичних систем криптоаналітиком 3-го рівня з поліноміальною чи суб'експоненційною складністю. Важливою особливістю постквантового періоду є значна невизначеність щодо вихідних даних для криптоаналізу та протидії в частині можливостей квантових комп'ютерів (КВК), їх математичного та програмного забезпечень, а також застосування для криптоаналізу існуючих криптоперетворень та криптопротоколів [1, 2]. Їх застосування необхідно розглядати щодо перехідного та постквантових періодів. Перехідний період пропонується визначити як проміжок часу у майбутньому, коли будуть суттєво вдосконалені класичні методи та засоби криптоаналізу, а також будуть створені та застосовуватись для криптоаналізу квантові комп'ютери з обмеженими потужностями. У цей період, можуть бути застосованими деякі існуючі нині стандарти асиметричних та симетричних криптографічних перетворень, але з максимально можливими чи збільшеними довжинами та властивостями загальносистемних параметрів та ключових даних, в тому числі при використанні в системі Блокчейн (БЧ). Постквантовий період пропонується визначити як проміжок часу у майбутньому, коли будуть суттєво удосконалені класичні методи та створені КВК з необхідними для успішного криптоаналізу довжинами регістрів (в кубітах) та необхідне для їх реалізації математичне та програмне забезпечення, особливо при сумісному застосуванні ІТ сумісно з БЧ.

Нині уже практично створені та застосовуються квантові комп'ютери. Стан створення та можливості застосування КВК є такими [1, 2]:

- IBM повідомила про план запуску в жовтні 2019 53 кубітного КВК;
- 53 – кубітний КВК IBM має нову конструкцію процесора, масштабуємий, знижена ймовірність помилок, надійний в хмарі;
- IBM відкриває новий обчислювальний центр в Нью-Йорку, 1 – 53 кубіт, 5 – 20 кубіт (14 в перспективі);
- 72 кубітний КВК Google за 3.5 хвилини виконує еквівалент роботи 10 тис. р. найпотужного кластера.

Стан створення асиметричних криптоперетворень типу електронний підпис (ЕП), асиметричний шифр (АСШ) та протокол інкапсуляції ключів (ПІК) наведено нижче [1, 2]:

- 2006 – 1ша PQC конференція у Льовен, Бельгія;
- 2009 – Дослідження NIST PQC Квантовостійка криптографія з відкритим ключем: опитування [Perlner, Cooper];
- 2012 – NIST починає проект PQC;
- Квітень 2015 – Семінар NIST з питань кібербезпеки в постквантовому світі;
- Серпень 2015 – Оголошення АНБ;
- Лютий 2016 – Звіт NIST про PQC (NISTIR 8105);
- Лютий 2016 – Оголошення NIST про "змагальний процес" у PQCrypto в Японії;
- Грудень 2016 – Опубліковані підсумкові вимоги та критерії оцінювання;
- Листопад 2017 – Кінцевий термін подання заявок;
- Грудень 2017 – Початок першого туру – 69 кандидатів, прийнято як "повні та вірні";
- Квітень 2018 – 1й семінар зі стандартизації NIST PQC;
- Січень 2019 – оголошено кандидатів 2 раунду;
- Серпень 2019 – 2й семінар зі стандартизації NIST PQC.

NIST США для оцінювання існуючих та перспективних проектів стандарту щодо ЕП, АСШ та ПІК запропоновані та уже використовуються такі рівні безпеки:

- I Рівень Принаймні так важко зламати, як AES128 (вичерпний перебір ключів);
- II Рівень Принаймні так важко зламати, як SHA256 (пошук колізії);
- III Рівень Принаймні так важко зламати, як AES192 (вичерпний перебір ключів);
- IV Рівень Принаймні так важко зламати, як SHA384 (пошук колізії);
- V Рівень Принаймні важко зламати, як AES256 (вичерпний перебір ключів).

NIST попросив заявників зосередитись на рівнях 1, 2 та 3 (рівні 4 і 5 призначені для дуже високої безпеки).

Продуктивність – вимірюється на різних класичних платформах.

Інші властивості: Змінні дані (параметри), ідеальна пряма секретність, простота та гнучкість, зловживання стійкістю та ін. стійкість до атак бічними каналами**.**

Метою доповіді є аналіз стану захищеності існуючих криптографічних перетворень та обґрунтування необхідності, стану розробки, дослідження та прийняття постквантових стандартів криптоперетворень, розгляд існуючої проблеми постквантової стандартизації та шляхів її вирішення на міжнародному та національному рівнях.

II. Сутність та стан вирішення проблеми постквантової криптографії на світовому рівні

NIST США провів 1-й етап конкуру щодо кандидатів на стандарти постквантових асиметричних криптографічних примітивів. Із 64 кандидатів першого етапу до другого рекомендовано 26. Суттєві досягнення зі створення математичних та програмних моделей перетворень на квантових комп'ютерах за оцінками провідних світових фахівців в галузі кібербезпеки, призвели до істотного прогресу в області криптографічного аналізу сучасних стандартизованих, особливо у постквантовий період. Вони зумовили стрімкий розвиток систем криптоаналізу.

Кандидатами на постквантові стандарти є симетричні шифри (СШ), АСШ (E2EE), ЕП та ПІК наведені нижче [1, 2]:

- симетричні блокові та потокові криптографічні перетворення (ДСТУ 7624:2014);
- криптографічні перетворення, що засновані на застосуванні геш-функцій (ДСТУ 7564:2014);
- асиметричні криптографічні перетворення на алгебраїчних решітках;
- криптографія, що заснована на кодах;
- криптографія, що заснована на складності обчислень ізогеній ЕК;
- багатовимірні криптоперетворення (мультиваріативні).

Основні вимоги до кандидатів на стандарти постквантових криптоперетворень можна конкретизувати у трьох таких напрямках:

- вимоги з безпеки (вимоги до стійкості до криптографічного аналізу);
- техніко-економічні вимоги (в основному щодо часової та просторової складностей);
- технічні характеристики реалізації алгоритмів асиметричних криптоперетворень.

Вимоги до стійкості мають бути сформульовані у відповідності до таких моделей загроз [1, 2]:

- для асиметричного шифрування – в умовах дії моделі IND-CCA2 (Indistinguishability under Adaptive Chosen Ciphertext Attack) – стійкість до атаки на основі адаптивно підібраного (вибраного) шифртексту;
- для електронного підпису – в умовах дії моделі EUF-CMA (Existentially unforgeable under adaptive chosen message attacks), тобто забезпечення захисту від екзістенційної підробки при атаках на основі адаптивно підібраного (вибраного) шифртексту;
- для протоколу обміну(встановлення, інкапсуляції) ключів – в умовах дії моделі безпеки Canetti-Krawczyk (СК-безпека).

Підсумки конкурсу за 1-й етап наведені в табл. 1. У ході виконання 1-го етапу в якості асиметричних постквантових механізмів рекомендовані механізми на основі: алгебраїчних решіток; математичних кодів (СВ-криптографія); мультиваріативного квадратичного перетворення; геш-функцій та ізогеній ЕК. Відповідні дані наведено в табл. 1 [1, 2].

ТАБЛИЦЯ 1

Підсумки 1-го етапу конкурсу

На основі решіток	5	21	26
На основі кодів	2	17	19
На основі мультиваріативних перетворень	7	2	9
На симетричній основі	3		3
Інші	2	5	7
Всього	19	45	64

У таблиці 2 наведено проміжні результати, що отримані в ході початкових досліджень на 2-му етапі

ТАБЛИЦЯ 2

Підсумки 2-го етапу конкурсу

На основі решіток	3	9	12
На основі кодів	0	7	7
На основі мультиваріативних перетворень	4	0	4
На симетричній основі	2		2
Інші	0	1	1
Всього	9	17	26

III. Аналіз стійкості існуючих кандидатів на постквантові стандарти

До основних задач, які можуть бути вирішені на квантовому комп'ютерові необхідно віднести такі [1]:

1) квантовий алгоритм факторизації Шора;
2) квантовий алгоритм Гровера пошуку елемента в несортованій базі;
3) квантовий алгоритм Шора для розв'язку дискретного логарифму в скінченому полі;
4) квантовий алгоритм розв'язку дискретного логарифму в групі точок ЕС Шора;
5) квантові алгоритми криптоаналізу для перетворень в фактор кільці;
6) квантовий алгоритм криптоаналізу Ксіонга та Ванга та його вдосконалення тощо.

Основними критеріями обмеження квантових атак є:

- 2^{40} логічних вентилів, тобто приблизної кількості вентилів, яка буде послідовно виконуватись за рік;
- 2^{64} логічних вентилів, яку сучасні класичні обчислювальні архітектури можуть виконувати послідовно за десять років;
- не більше, ніж 2^{96} логічних вентилів, тобто приблизна кількість вентилів як кубіти атомного масштабу зі швидкістю світла часу поширення може виконувати за тисячоліття.

Криптографічна стійкість перспективних проектів повинна бути такою:

1) Стійкість проти класичних атак – Класична безпека.

2) Стійкість проти «квантових» атак. Зокрема, стійкість до алгоритму Гровера (необхідність подвоєння розміру ключа) – Квантова безпека.

3) Базування на задачах, які мають високу складність обчислення. Можливе ігнорування зниження рівня складності, за умови, що практична стійкість не зміниться – Доказова безпека.

4) Можливість використання у протоколах типу TLS 1.3 з підтримкою forward secure cipher suites – Довгострокова безпека.

5) Стійкість проти атак з адаптивним підбором – Активна безпека.

Повинна бути забезпеченою стійкість проти атак спеціального виду. Реалізація цих атак направлена на пошук вразливостей у практичній реалізації криптосистеми, в першу чергу засобу КЗІ:

- контроль над обчислювальним процесом;
- спосіб доступу до системи чи засобу;
- метод безпосереднього здійснення атаки тощо.

В основу захисту від атак спеціального виду можуть бути покладеніособливості:

- фіксована кількість звернень до геш-функції;
- рандомізація даних;
- незалежність ключів від значень тощо.

Результати порівняльного аналізу складності факторизації для класичного та квантового алгоритмів наведені в таблиці 3.

ТАБЛИЦЯ 3

Результати порівняльного аналізу складності факторизації

Розмір модуля N, бітів	Кількість необхідний кубітів, $2n$	Складність квантового алгоритму, $4n^3$	Складність класичного алгоритму
512	1024	$0{,}54\cdot10^{9}$	$1{,}6\cdot10^{19}$
3072	6144	$12\cdot10^{10}$	$5\cdot10^{41}$
15360	30720	$1{,}5\cdot10^{13}$	$9{,}2\cdot10^{80}$

Результати порівняльного аналізу складності дискретного логарифмування для класичного та квантового алгоритмів наведені в таблиці 4.

ТАБЛИЦЯ 4

Результати порівняльного аналізу складності дискретного логарифмування

Алгоритм розв'язку дискретного логарифмічного рівняння			
Розмір порядку базової точки, бітів	Кількість необхідний кубітів $f(n)=7n+4\log_2 n+10$	Складність квантового алгоритму $360n^3$	Складність класичного алгоритму
163	1210	$1{,}6\cdot10^{9}$	$3{,}4\cdot10^{24}$
256	1834	$6\cdot10^{9}$	$3{,}4\cdot10^{38}$
571	4016	$6{,}7\cdot10^{10}$	$8{,}8\cdot10^{85}$
1024	7218	$3{,}8\cdot10^{11}$	$1{,}3\cdot10^{154}$

Основними завданнями 2-го етапу є такі (NIST ще відкритий для злиття заявлених проектів):

- Нові оператори IP потрібні тільки в разі приєднання нових членів команди або зміни статусу IP;
- Згодом питання, пов'язані з IP, можуть зіграти більшу роль у прийнятті наших рішень;
- 2-й раунд триватиме 12-18 місяців, після чого ми очікуємо, що відбудеться 3-й раунд;
- Загальна хронологія: NIST ще очікує проекти стандартів близько 2022 року (але залишає за собою право змінити це!).

IV. Стан створення постквантових стандартів на національному та міжнародному рівнях

В Україні розроблені та рекомендовані до застосування постквантові стандарти таких симетричних криптоперетворень.

1. Алгоритм блокового симетричного перетворення ДСТУ 7624:2014 – 10 режимів роботи (128, 256, 512біт) (5-7 рів.).

2. Функція гешування ДСТУ 7564:2014 (8–512 біт) (5-7 рів.).

3. Алгоритм симетричного потокового перетворення ДСТУ 8845:2019 (256–512 біт) (5-7 рівень).

4. На стадії розгляду першої редакції знаходиться проект стандарту «Алгоритми асиметричного шифрування та інкапсуляції ключів» (алгебраїчні решітки, стійкість 256, 384, 512 біт).

5. Розроблено проект та на стадії досліджень знаходиться проект стандарту «Електронний підпис» (алгебраїчні решітки, стійкість 256, 384, 512 біт).

Постквантові кандидати на ЕП за 2-й етап наведено в таблиці 5 [1, 2].

ТАБЛИЦЯ 5

Постквантові кандидати на ЕП за 2-й етап

Математичні методи ЦП	Кандидати в стандарти ЦП	Число
Алгебраїчні решітки	Crystals-Dilithium, Falcon, qTesla	3
Геш-перетворення	Sphincs+	1
Кінцеві автомати	Picnic	1
Мультиваріативні перетворення	Gemss, Luov, Mqdss, Rainbow	4
Усього		9

Механізми постквантових кандидатів на АСШ та ПІК за 2-й етап наведено в таблиці 6 [1, 2].

ТАБЛИЦЯ 6

Постквантові кандидати на АСШ та ПІК за 2-й етап

Математичні методи ПІК	Кандидати в стандарти ПІК	Число
Алгебраїчні коди	BIKE, Classic McEliece, HQC, LEDAcrypt (LEDAkem, LEDApkc), NTS-KEM, Rollo (Locker, Ouroboros-r(lattice)), RQC	7
Алгебраїчні решітки	Crystals-Kyber, FrodoKEM, LAC, NewHope, NTRU (NTRUEncrypt, NTRU-HRSS-KEM), NTRU Prime, Round5 (Hila5, Round2), Saber, Three bears	9
Інші перетворення	SIKE (ізогенії)	1
Усього кандидатів		17

V. Сутність методу асиметричного шифрування та інкапсуляції ключів

В якості основного математичного методу постквантового АСШ та ПІК вибрано NTRU-подібні криптоперетворення. Нині вони отримали назву як таких, що ґрунтуються на алгебраїчних решітках. Вказана назва пов'язана з тим, що доведення їх криптографічної стійкості зводиться до застосування математичного апарату алгебраїчних решіток, а також, по суті, NTRU криптографічне перетворення є безпосередньо алгебраїчною решіткою.

Проведені дослідження підтвердили можливість використання цих параметрів для шифрування з боку дуже важливої характеристики – швидкодії.

Зважаючи на національні вимоги, можливості та реалізацію національного стандарту симетричного шифрування ДСТУ 7624-2014, стосовно проекту національного АСШ висунуто вимоги, що будь-яка атака, що порушує певне визначення безпеки застосування АСШ, повинна вимагати обчислювальні ресурси порівняні з, або більші, ніж необхідні для пошуку 512-бітного ключа (наприклад, Калина-512) стосовно класичної атаки та 256-бітного ключа стосовно квантової атаки.

Додатковими властивостями щодо криптографічної стійкості АСШ є наступні.

Повна пряма захищеність. Цей термін вживається для позначення характеристик ключа у протоколах узгодження, який надає гарантії, що попередній сеансовий ключ не буде скомпрометований, якщо навіть секретний ключ серверу був скомпрометований. Хоча ця властивість виконується при стандартному шифруванні, вартість такої операції може бути недозволеною в деяких випадках.

Ще одним випадком взаємовпливу захищеності та швидкодії є здатність протидії атакам по стороннім каналам. Більш бажаними є схеми, що здатні протидіяти цим атакам при мінімальній вартості.

Третьою бажаною властивістю є протидія мультиключовим атакам. В ідеалі атакуючий не повинен відчувати різницю, коли його мета – скомпрометувати одну пару ключів або велику кількість пар ключів.

Остання вимога, недостатньо добре визначена, – це стійкість до неправильного використання механізму АСШ. Бажано щоби АСШ не міг ставати непрацездатними через окремі помилки в коді, несправності у генераторі випадкових чисел.

Остання вимога, недостатньо добре визначена, – це стійкість до неправильного використання механізму АСШ. Бажано щоби АСШ не міг ставати непрацездатними через окремі помилки в коді, несправності у генераторі випадкових чисел.

У таблиці 7 наведені ідентифікатори та характеристики механізмів (алгоритмів) АСШ, що обрані для їх порівняння, при чому швидкодія $T_{пр.}$ та $T_{зв.}$ задана в мільйонах (10^6) тактів [1] роботи комп'ютера.

При дослідженні з метою вибору кращих алгоритмів також були висунуті додаткові безумовні вимоги [3]:

1) алгоритм повинен гарантувати, щонайменше 5 рівень безпеки за класифікацією NIST;

ТАБЛИЦЯ 7

Характеристики алгоритмів шифрування

Алгоритми	Тип	$I_{ст.}$	$l_{в.к}$	$l_{о.к}$	$l_{рез.}$	$T_{пр.}$	$T_{зв.}$
Giophantus	Lattices	5	27204	1134	54408	420,543208	792,577
KINDI	Lattices	5	2368	2752	3328	0,705	0,919
LAC	Lattices	5	1056	2080	2048	0,137258	0,133
LEDApkc	Codes	5	12384	40	24768	92,84	264,938
LIMA	Lattices	5	12289	18433	12291	0,909	1,126
Lizard	Lattices	5	8192	998266	8512	0,805	0,568
LOTUS	Lattices	5	1470976	1630720	1768	0,901	1,087
McNie	Codes	5	630	584	729	3,504	7,707
NTRUEncrypt	Lattices	5	64232	63912	127696	174,2	0,299
Odd Manhattan	Lattices	5	4454241	4456650	616704	141,625	155,302
OKCN/AKCN/CNKE	Lattices	5	1312	992	1200	0,568	0,631
Round2	Lattices	5	830	1039	953	0,905	1,135
RQC	Codes	5	40	1795	3574	6,46	18
Titanium	Lattices	5	23552	32	8320	2,974	0,561
NTRU Prime Ukraine	Lattices	5	1578	243	1578	0,074	0,138

2) якщо існує декілька варіантів наборів параметрів для одного алгоритму, то в порівнянні бере участь варіант, що гарантує найбільшу безпеку.

3) Перспективний проект стандарту АСШ повинен також, при необхідності, забезпечувати 512 бітну класичну криптографічну стійкість та відповідно 256 бітну квантову криптографічну стійкість.

На рис. 1 відображено гістограму відносної переваги алгоритмів. Як видно найбільшу перевагу має алгоритм NTRU Prime Ukraine, на другому місці – LAC, на третьому – OKCN/AKCN/CNKE.

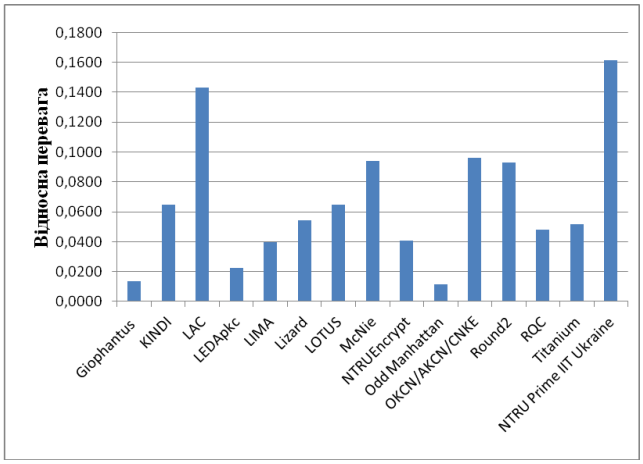


Рисунок 1 – Відносна перевага алгоритмів шифрування

VI. Основні властивості національного постквантового стандарту (перша редакція) «Скеля»

У даному розділі наведено основні характеристики та властивості національного постквантового стандарту «Скеля» [4].

У таблиці 8 наведено основні загальні параметри АСШ «Скеля».

У таблиці 9 наведено основні параметри для генерації ключа «Скеля».

ТАБЛИЦЯ 8

Основні загальні параметри АСШ «Скеля»

Позн.	Призначення	Формула
$(Z/q)[x]$	Кільце поліномів Кожен елемент Z/q зазвичай кодується у $\lceil \log_2 q \rceil$ біт	
n	Порядок поліному. Визначає кількість його коефіцієнтів. Просте число, для якого поліном $x^n - x - 1$ є незвідним	$n \geq \max\{3, 2t\}$
R	Поле поліномів $Z[x]$ з модулем $x^n - x - 1$	$Z[x]/(x^n - x - 1)$
$R/3$	Поле поліномів $(Z/3)[x]$ з модулем $x^n - x - 1$	$(Z/3)[x]/(x^n - x - 1)$
R/q	Поле поліномів $(Z/q)[x]$ з модулем $x^n - x - 1$	$(Z/q)[x]/(x^n - x - 1)$
p	Менший модуль	$p = 3$
q	Більший модуль, просте число, за яким зводяться усі коефіцієнти поліному R/q	$q \geq 48t + 3$
t	Натуральне число, кількість ненульових елементів поліному залежить від цього параметру.	$t \geq 1$
k	Рівень криптостійкості	128, 256, 512
m	Таємне повідомлення. Кількість 0, 1 та -1 не менше, ніж t.	$m \in R/3$
M	Повідомлення після доповнення випадкового рядка b та іншої інформації.	
octL	Поле для завдання довжини повідомлення	1 байт
e	Зашифроване повідомлення	$e \in R/q$

ТАБЛИЦЯ 9

Основні параметри для генерації ключа «Скеля»

Позн.	Призначення	Формула
G	Випадковий t-малий елемент (поліном), оборотний у $R/3$. Кількість ненульових елементів дорівнює 2n/3+1 Секретний параметр, що використовується для обчислення відкритого ключа.	$g \in R/3$
F	Випадковий t-малий елемент (поліном). Кількість ненульових елементів дорівнює 2t. Секретний параметр, що використовується для обчислення відкритого ключа.	
f	Малий елемент (поліном), незвідний в R/q, є секретним (особистим) ключем. $f=p*F+1$	$f = (1+3F) \bmod q$ $f \in R/q$
h	Відкритий ключ відправника. Оборотний елемент у R/q.	$h = p * G/f$
$\underline{h}$	h перетворене в рядок октетів. Довжина $\underline{h}$ дорівнює $n\lceil \log_2 q \rceil$	

У таблиці 10 наведено додаткові параметри АСШ «Скеля».

ТАБЛИЦЯ 10

Додаткові параметри NTRUPrime Ukraine «Скеля»

Позн.	Призначення	Формула
qBits	Кількість біт q	$qBits = \lceil \log_2 q \rceil$
r	Засліплюючий поліном, випадковий t-малий елемент.	$r \in R/3$
b	Випадковий компонент (salt), що доповнює повідомлення.	
db	Довжина випадкового компонента в бітах. Визначається рівнем криптостійкості.	$db = \begin{cases} 256 & k \leq 256 \\ 512 & k \leq 512 \end{cases}$ $db \bmod 8 \equiv 0$
Buffer Len Bits	Довжина бітового рядка для його перетворення в малий поліном. Використовується при зашифруванні	$\lfloor (((n-1)/2) \cdot 3)/8 \rfloor \cdot 8$
maxMsgLenBytes	Максимальна довжина повідомлення для зашифрування.	*bufferLenBits/8 – hLenBytes - 1*
c	Кількість бітів, яка використовується для визначення індексу ненульового елементу полінома	$c_1 = \lfloor log_2 n \rfloor$; $c_2 = c_1 + 1$ $c_3 = c_2 + 1$; $c = c_i + 1$ для якого $c_i \bmod n$ мінімально
Llen	Кількість байтів для завдання довжини повідомлення, що шифрується	$\log_{256}$ maxMsgLenBytes
tm	Кількість 1(-1) в поліномі, який створюється після перетворення повідомлення, що шифрується	$tm=t$
Hash	Геш-функція, що використовується	Залежить від криптостійкості
Hlen	Довжина геш-значення	$Hlen=k$

Min CallsR	Мінімальна кількість викликів функції гешування для створення полінома для осліплення	$\lceil t \cdot c / hashLenBits \rceil$
Min Calls Mask	Мінімальна кількість викликів функції гешування для маскування повідомлення	$\lceil \lceil n / 5 \rceil \cdot 1.5 / hLenBits \rceil$
pkLen	Кількість бітів відкритого ключа, які використовуються при формуванні рядка для шифрування	pkLen=db
OID	Ідентифікатор методу. 3 байти	$OID[0] = 0; OID[1] = 1; OID[2] = 2;$

У таблиці 11 наведено часові характеристики функцій зашифрування та розшифрування АСШ «Скеля».

ТАБЛИЦЯ 11

Часові характеристики функцій зашифрування та розшифрування

N	q	Encrypt			Decrypt			Decrypt	
		Hash	Salsa20	Snow20	Hash	Salsa20	Snow20	Decrypt	Check
439	6833	56908	40832	36308	81356	64280	60016	31708	29848
457	6037	64012	38200	34384	87948	61448	56868	30500	28496
461	7607	70592	42080	37724	99140	68572	63420	32524	32476
461	8779	70396	41964	37724	98352	68100	63220	32648	31928
467	3911	52828	32080	29592	72760	51528	48532	26908	23320
463	6529	65364	39124	35764	90484	63256	59072	32352	29220
463	6841	67780	40628	36500	94884	66448	61464	32336	30884
463	9371	70832	42824	38316	97344	68056	62296	33644	32756
479	5689	63824	38432	34144	87748	61556	57320	31000	28360
479	6089	65548	39072	35440	90624	62596	58352	32168	28948
491	6287	69464	41572	36188	94220	65984	61500	31368	30600
761	4591	74584	43640	41616	102232	71004	68984	40536	31820

У таблиці 12 наведено обчислювальну складність АСШ «Скеля».

ТАБЛИЦЯ 12

Обчислювальна складність АСШ «Скеля»

Режим АСШ (Скеля)	Скеля 512 (SHA 256) Мбіт/с	Скеля 256 (SHA 256) Мбіт/с
Encrypt	43	47
Decrypt	23	29

Режим АСШ (Скеля)	Скеля 512 (Купина 512) Мбіт	Скеля 256 (Купина 256) Мбіт
Encrypt	35	40
Decrypt	20	26

Алгоритми зашифровування «Скелі 512» (функція encrypt)

Параметри:

n – порядок поліному, визначає кількість його коефіцієнтів. Просте число, для якого поліном є незвідним; просте число;

t – натуральне число, яке визначає кількість ненульових елементів малого поліному;

p – менший модуль (p=3) для цього стандарту;

q – більший модуль (просте число);

maxMsgLenBytes – Максимальна довжина повідомлення;

db – довжина випадкового двійкового рядка;

bufferLenBits – довжина буферу для доповненого повідомлення;

octL – довжина кількості октетів для повідомлення визначається параметром octL;

OID – ідентифікатор методу;

minCallsR – кількість викликів функції гешування при формуванні засліплюючого поліному;

minCallsMask – кількість викликів функції гешування при формуванні маскуючого поліному.

Компоненти:

Генератор випадкових чисел (rand);

Функція RE2BSP – перетворення коефіцієнтів полінома в бітовий рядок;

Функція RE2OSP – перетворення коефіцієнтів поліному в рядок октетів;

Функція GenBP – генерація засліплюючого поліному;

Функція MGF – генерація маски.

Вхід:

m – повідомлення для за шифрування (рядок октетів);

mLenBytes – довжина рядка m (кількість октетів);

h – відкритий ключ (більший поліном та відповідний бітовий рядок).

Для обчислення шифртексту необхідно виконати такі або еквівалентні кроки.

1. Перевірка довжини повідомлення.

Якщо mLenBytes > maxMsgLenBytes То

RetValue = ERROR e=0;

2. Формування випадкового рядка b довжиною

3. Формування рядка октетів M довжини *bufferLenBits* .

4. Перетворення рядка октетів M в поліном MTrin за допомогою кодування 3-х бітових рядків в 2 коефіцієнта меншого полінома.

5. Формування рядка октетів sData, який складається з:

OID – ідентифікатор методу;

Повідомлення для за шифрування m;

випадковий рядок b довжиною db бітів (db mod 8 = 0);

байтів відкритого ключа h, перетвореного в рядок байтів за допомогою стандартного компонента для перетворення полінома в рядок бітів RE2BSP, з цього рядка використовуються перші *pkLen бітів*, яка позначена hTrans. Довжина рядка sDataLen=3+*mLenBytes+2*db/8.*

6 Формування засліплюючого полінома r з використанням рядка октетів sData та його довжини r:=BPGM (sData, sDataLen).

7 Обчислення R:=r *h в полі (Z/qZ)[X](X^n–X–1).

8 Обчислення R4:=R mod 4.

9 Перетворення R4 в рядок октетів (стандартний компонент RE2OSP). Отримання oR4.

10 Генерація полінома для маскування mask (функція MGF) звикористанням рядка oR4 в якості seed.

11 Обчислення поліному *m'*=(M+mask) mod p.

12 Якщо кількість 1, -1, 0 в поліномі *m'* менше, ніж *tm* то перейти на крок 1.

13 Обчислення шифротексту: *e*:=*R*+*m'* mod *q* (більший поліном).

Перетворення *e* в рядок октетів E (функція RQOS).

Значення, що повертається:

RetValue=OK

шифртекст E (Рядок октетів)

Алгоритми розшифровування «Скеля-512» (функція decrypt)»

Параметри

n – порядок поліному, визначає кількість його коефіцієнтів. Просте число, для якого поліном є незвідним; просте число

t – натуральне число, яке визначає кількість ненульових елементів малого поліному

p –менший модуль (p=3) для цього стандарту

q – більший модуль (просте число)

maxMsgLenBytes – Максимальна довжина повідомлення

db – довжина випадкового двійкового рядка

bufferLenBits – довжина буферу для доповненого повідомлення

octL – довжина кількості октетів для повідомлення визначається параметром octL

OID – ідентифікатор методу

Компоненти:

Функція OS2BSP – перетворення рядка октетів в рядок бітів

Функція RE2BSP – перетворення коефіцієнтів полінома в бітовий рядок

Функція RQOS – перетворення коефіцієнтів полінома в рядок октетів

Функція OSRQ – перетворення рядка октетів E в поліном

Функція GenBP – генерація засліплюю чого поліному

Функція MGF – генерація маски

Вхід:

E –шифртекст (рядок октетів);

mLenBytes – довжина рядка m (кількість октетів)

h – відкритий ключ (більший поліном та відповідний бітовий рядок)

Значення, що повертаються:

1 – Признак успішності (Retvalue=OK, ERROR)

2 – m (розшифроване повідомлення, рядок октетів) в разі успішного завершення

3 – mlen – довжина розшифрованого повідомлення – октетів (разі успішного завершення)

Для обчислення відкритого повідомлення необхідно виконати такі або еквівалентні кроки.

1. Перетворення рядка октетів E в поліном *e* (R/q) – функція OSRQ.

2. Обчислення a:= f * e в полі (Z/qZ)[X](X^n–X–1)

3. Обчислення cm' = a mod p

4. Якщо кількість 1, -1, 0 в поліномі *cm'* менше ніж *tm* то

RetValue=Error

m=mlen=0;

Перейти на крок 18.

5. Обчислення cR:=e – m' (mod q) (кандидат на r*h)

6. R4 := cR mod 4

7. Перетворення R4 в рядок октетів (функція RQOS для q=4). Отримання coR4

8. Генерація полінома для маскування mask (функція MGF)

9. Обчислення меншого поліному cmTrin:=cm'–mask (mod p)

10. Перетворення cmTrin в бітовий рядок cM згідно заміни двох коефіцієнтів трьома бітами (табл. 7.1). Якщо для перетворення треба коефіцієнти полінома -1, -1 перетворити в бітовий рядок, то

RetValue = Error

m = mlen = 0;

Перейти на крок 18.

11. Якщо довжина отриманого рядка l mod 8!=0 то видалення з бітового рядка останніх l mod 8 нулів

12. Перетворення бітового рядка Mbin в рядок октетів M (функція BS2OSP)

13. Розібрати рядок октетів M:

13.1 Перші bLen октетів – рядок b

13.2 Наступний байт – довжина повідомлення lm; Якщо lm>maxMsgLenBytes, то

RetValue = Error

m = mlen = 0;

Перейти на крок 18

13.3 Наступні lm байтів – повідомлення m

13.4 Наступні байти – нульові. Якщо ці байти відрізняються від нульових, то

RetValue = Error

m = mlen = 0; Перейти на крок 18

14. Значення, що повертаються: RetValue, m, mlen

Режими роботи

Механізм (алгоритм) асиметричного блокового криптографічного перетворення реалізовано на основі виконання перетворень в кільцях поліномів та скінчених полях. У залежності від довжин (розмірів) загальних параметрів та асиметричних пар ключів (особистого (секретного) та відкритого ключів), застосовуються два режими роботи:

- режим 256 біт захищеності від класичної атаки та 128 біт захищеності від квантової атаки, що позначається як Скеля256/128;
- режим 512 біт захищеності від класичної атаки та 256 біт захищеності від квантової атаки, що позначається як Скеля512/256.

В кожному із режимів роботи за рахунок застосування криптографічних перетворень в кільцях поліномів та скінчених полях забезпечується надання послуг асиметричного шифрування та неспростовності отримувача.

В кожному із вказаних режимів роботи забезпечується захист від атак сторонніми каналами (через витік по технічним каналам).

Висновки

1. Основною особливістю вимог щодо розроблення проекту національного стандарту АСШ ПІК є забезпечення крім 5-го рівня безпеки, також 7-го рівня – 512 біт класичної криптографічної стійкості та відповідно 256 біт квантової криптографічної стійкості. Причому під 6 рівнем безпеки будемо розуміти забезпечення 384 біт безпеки щодо класичних атак та 192 біт щодо квантових атак. Вказана вимога визначена для всіх складових механізму АСШ ПІК.

2. У криптосистемі «NTRU Prime IIT Ukraine» в якості основного криптоперетворення, як в NTRU Prime, на відміну від NTRU, застосовується перетворення в скінченому полі. Вказане унеможливлює проведення щодо криптографічної системи «NTRU Prime IIT Ukraine» ряду потенційних атак та виключає потенційні слабкості, що присутні криптосистемі NTRU. В основному вони пов’язані з існуванням нетривіальних підкілець чи фактор кілець фактор кільця (зрізаних) поліномів.

3. У криптосистемі «NTRU Prime IIT Ukraine» поліноми F та r є довільними t-малими, вони мають $2t$ ненульових коефіцієнтів (+1, -1) в той час як в NTRU кожен з зазначених поліномів має точно t ненульових коефіцієнтів, які дорівнюють 1 та -1 відповідно. Аналогічне справедливе і для полінома g, який використовується у криптосистемі «NTRU Prime IIT Ukraine», – він є довільним малим поліномом з $2t$ ненульовими коефіцієнтами (+1, -1). Вказане дозволяє розширити у порівнянні з NTRU розмір ключового простору без втрати ефективності реалізації алгоритмів формування ключів та виконання алгоритмів зашифрування і розшифрування.

Література

[1]. Post-Quantum Cryptography. – Electronic resource. – Access mode: https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

[2]. Post-Quantum Cryptography. Workshops and Timeline. – Electronic resource. – Access mode: https://csrc.nist.gov/Projects/post-quantum-cryptography/workshops-and-timeline.

[3]. Yesina Maryna, Olga Akolzina, Olena Kachko (supervisor). Proposals of the expert estimations technique usage for the comparing and estimation NTRU-like cryptographic systems // Inżynier XXI wieku (“Engineer of XXI Century” – the VII Inter University Conference of Students, PhD Students and Young Scientists: University of Bielsko-Biala, Poland, December 08, 2017). – Bielsko-Biała: Wydawnictwo Naukowe Akademii Techniczno-Humanistycznej w Bielsku-Białej, 2017. – P. 383–398. – ISBN 978-83-65182-81-4 (Tom 2) – Chapter in monograph. (Розділ в монографії).

[4]. Gorbenko I. D. General statements and analysis of the end-to-end encryption algorithm NRTU Prime IIT Ukraine / I. D. Gorbenko, O. G. Kachko, M. V. Yesina // Радиотехника. – Х. : Харьковский национальный университет радиоэлектроники, 2018. – Выпуск 193 – С. 5–16.

Quantum Information Paradigm of Physical Reality

Eugene Machusky[1]

1. Department of Physics and Information Security Systems, National technical university of Ukraine, UKRAINE, Kyiv, Prosp.Peremohy, 37,
E-mail: sivera@ukr.net

The unified information-entropy-energy paradigm of quantum physics, quantum chemistry and quantum computing is logically developed and substantiated. The analytically derived matrix equations of mutual functional relations of transfinite numbers pi and e with infinite progressive and inverse natural sets fully describe the dynamic geometry of sub-atomic and hyper-atomic space, create a quantum bridge between discrete and continuous mathematics, as well as between special and general theory of relativity, as well as between the physical theories of strings, super symmetry, loop gravity, and multiverse. The median values and entropy of fundamental quantum units of standard physical model are calculated with an accuracy, which is limited only by the bit capacity of the computing device. It is shown that new metric system SI-2019 and modern theories of quantum physics have serious logical fallacies restricting an accuracy of mutual coordination of metric units and bringing an excessive errors to quantum technology, quantum communications and quantum computing.

I. Introduction

During the past 100 years, many scientists have been amazed at the remarkable mathematical efficiency of quantum physics. But the simplest and perhaps the only reason for this is that quantum physics, by its nature and origin, is actually no more and no less than the universal metrical and universal computing system based on well-measured values and decimal orders of energy quanta. This can be clear illustrated by the seven parametric equations, which functionally connect the transcendental numbers pi and e with an infinite set of integers and their inverse values:

[Ri] = 1+2/100*(e+[Ai]*(1+Sqrt(2*pi*e/100))) - matrix of relative inverse radius of pulsating sphere

[Pi] = 2*pi*[Ri] - matrix of relative inverse perimeter of pulsating sphere

[Gi] = [Pi]*(1+[Ai]) - matrix of density of relative inverse perimeters of pulsating sphere

[Ci] = [Ri]^64*10^7 - matrix of relative stroboscopic speed of pulsations

[Mi] = 12-[Ai]/10 - matrix of relative amplitude of pulsations

[Ki] = Cos[Mi]-Sin[Mi] - matrix of relative phase of pulsations

[Ni] = 100*(Sqrt(8*pi*e/(8*pi*e+137^2))/(1+2* *[Ai]/1000)-5/10^8) - matrix of relative entropy of eccentricity

All parameters of the presented equations are functions of the argument of informational entropy Sqrt(2*pi*e) of the Gaussian normal distribution. Despite its amazing simplicity, the equations completely integrate the five sections of mathematics (arithmetic, geometry, algebra, logarithms and big data statistics) into a universal computing system, creating a radial web and drawing a spherical wave interference pattern that is physically a photon hologram of subatomic space.

The Encyclopaedia Britannica defines mathematics as "the science of structure, order, and relationships arising from the elementary practice of counting, measuring, and describing the shapes of objects". Physics is defined as "a science that deals with the structure of matter and the interactions between the fundamental components of the observable universe. In a broad sense, physics deals with all aspects of nature at both the macroscopic and submicroscopic levels".

The matrices presented below combine the continuous and discrete mathematics with the main sections of physics (thermodynamics, electrodynamics, chromodynamics, gravidynamics) in the framework of universal metric of relative space-time:

Matrix of three-dimensional distribution of relative inverse eccentricity of pulsating sphere, where A = 137, B = 602214183, produces the set of Avogadro units [Ni] and describes the average energy entropy of ideal crystal state of matter:

A4 = 4/A-3*A0

AH = 1/16/pi/e

AL = 1/(1+59*Ln(10))

Median{AH...A4} = AH4

Median{AHL...AH4) = AE >>> NE...NA...NB <<< B/(1+4*pi/10^8)/10^8

Median{AH...AL} = AHL

Matrix of two-dimensional distribution of relative inverse perimeter of pulsating sphere produces the set of Planck units [Pi]

and describes the average energy entropy of ideal liquid state of matter:

A1 = 1/A

A0 = (pi*e/100)^2

AS = 1/100/(10/(10-1))^3

Median{A0...A1} = A01

Median{A0S...A01} = AP >>> PP...PQ...PF <<< AF = 1/(A+36/1000)

Median{A0...AS} = A0S

Matrix of one-dimensional distribution of relative inverse radius of pulsating sphere, R = Integer{10^8*(C/10^7)^(1/64)};

C = ((R+4*pi*C/10^10)/10^8)^64*10^7; K = e+AS+BS; BS = Sum{B/10^(3*N+8)} produces the speed units C, V, T and describes the average energy entropy of ideal gaseous state of matter:

RE = (R+1/e)/10^8

Median{RA...RE} = RAE

RA = (R+1/(e+AS))/10^8 Median{RAK...RAE} = RT >>> T...V...C <<< RC = (C/10^7)^(1/64)

Median{RA...RK} = RAK

RK = (R+1/K)/10^8

Matrix of zero-dimensional centralized distribution of the relative density of relative inverse perimeters of spherical waves,

where X = Root {X*e^X/(e^X-1) = 5}, produces the set of Newtonian gravitational units [Gi] in ideal crystal state GN, in liquid crystal state GQN, in ideal liquid state GQ, in liquid-gas state GQV, in ideal gas state GV, in ideal gas-vacuum state GVN, in vacuum state GX and in harmonic median state G.

GN = PN*(1+AN)

Median{GQ...GN} = GQN

GQ = PQ*(1+AQ) Median{GQV...GQN} = GVN...G...GX <<< AX = 5/X-1

Median{GQ...GV} = GQV

GV = PV*(1+AV)

Analytically determined and then calculated with an accuracy of 10^64 matrices of information entropy of Avogadro units [Ni], Planck units [Pi], Maxwell velocity units [Ci] are presented below:

N4 = 6.022141002581922659545427906392640549106329714296253865736376230 0
NH = 6.022141005390288386746703596666428308193753851391157315970739480 0
NB = 6.022141073235433817688323720807815457893918180548564351884739447 8
NL = 6.022141145015173011231009667252917868659482423130928517294158130 0

P1 = 6.626071005575500527563230680857474700731003678085499214729143248 8
PF = 6.626070665023663032530912906931917928683576215986747291241337514 9
P0 = 6.626069839825457876034665455901235682153693420703149998608674304 6
PS = 6.626069359237049533925846040470488611171283886026360267351777079 9

C = 299792457.867591338433683989149905009273372586654059140405331146 33
CE = 299792456.257274188286886027303032761337552565628547217370703488 39
CA = 299792456.078254512807124830945275462965319414254603079958988053 33
CK = 299792455.930943197787057254994665628647917051398787082513873446 93

The presented set of basic wave units enables us to eliminate SI metric artifacts and exchange Feynman energy diagrams for information diagrams of quantum functional analysis [1], [2], [3].

II. Information entropy of standard functional analysis

Standard functional analysis is based on transcendental numbers pi and e, which can be considered as spatial and temporal parameters of the harmonic wave motion. The computational entropy of functional analysis is determined by the arithmetic used (floating point, fixed point, or excluded point) and can reach many decimal orders depending on the digital length of the records of rational and irrational numbers. This can be easily illustrated by the following examples:

Computing of median value of pi and e with fixed decimal point

pi = 3/10^0+1/10^1+4/10^2+1/10^3+5/10^4+9/10^5+2/10^6+6/10^6+... = [3.1415926535897932]
e = 2/10^0+7/10^1+1/10^2+8/10^3+2/10^4+8/10^5+1/10^6+8/10^8+... = [2.7182818284590452]

Sqrt((pi^2+e^2)/2) = [2.937572169327930600000000000000000000000000000000000000000000000000000000000000] = mR
(pi+e)/2 = [2.929937241024419200000000000000000000000000000000000000000000000000000000000000] = mA
[mR+mA+mG+mH]/4 = [2.926109816185223442711454376999454189651465145760654843953502997 6] = med
Sqrt(pi*e) = [2.922282365322277827645817507997816758605860583042619375814011990 5] = mG
Mea 2/(1/pi+1/e) = [2.914647489066266143200000000000000000000000000000000000000000000000000000000000] = mH

Computing of median value of pi and e with excluded decimal point

PI = 3*10^N+1*10^(N-1)+4*10^(N-2)+1*10^(N-3)+5*10^(N-4)+9*10^(N-5)+2*10^(N-6)+... = [31415926535897932]

E = 2*10^N+7*10^(N-1)+1*10^(N-2)+8*10^(N-3)+2*10^(N-4)+8*10^(N-5)+1*10^(N-6)+... = [27182818284590452]

MR = [29375721693279306.063189584866066200000]
MA = [29299372410244192.000000000000000000000]
Med = [29261098161852234.442979775108485260375]
MG = [29222823653222778.276458175079978200000]
MH = [29146474890662661.432271340487896641500]

Anyway, we can not calculate the median with accuracy more than 18 decimal orders. The "trivial zeros" obtained in the mantissa of mean values are the main cause of the so-called "ultraviolet" and "infrared" catastrophes of the quantum physics. The elimination of such "catastrophes" requires “honest” computational algorithms working with interconnected quantum units, the decimal order of which can be very, very large, for example, for the Avogadro constant 10^(+23), or very, very small, as for Planck constant 10^(-34).

III. Fundamental units of quantum metric

Quantum physics in its origin is a natural synthesis of quantum metric and quantum calculus. The seven basic units of quantum metric were firstly derived in reference [1], and are identified here by the names of first investigators:

C = [299792457.867591338433683989149905009273372586654059140405331146 33] - constant of Maxwell
K = [2.731599998459045235360287471352662497757247093699959574966967627 7] - constant of Kelvin

A = 1/Lim{Sum{729927/10^(8*N)}} = 137 - integer of Sommerfeld
AS = Lim{Sum{[A+(A-100)*N]/10^(3*N+2)}} = 1/100/(10/(10-1))^3 = 729/10^5 - ratio of Schrodinger
B = 602214183 - integer of Avogadro
BS = Lim{Sum{B/10^(3*N+8)}}= 602817/10^8 - ratio of Dalton
R = Integer{10^8*(C/10^7)^(1/64)} = 105456978 - integer of Dirac

The constant C can be considered as the upper stroboscopic limit of the speed of harmonic circular motion. In a decimal positional system it can be calculated with an extreme accuracy of 10^(-64). The constant K can be considered as the upper stroboscopic limit of the speed of harmonic radial pulsation of sphere, it should also be estimated using the relative accuracy of C. With the same accuracy, we must use numbers pi and e when evaluating the parameters of wave motion:

Integer{pi*10^64}/10^64 = 3.14159265358979323846264338327950288419716939937510582097494459 23 = PI
C = 299792457.867591338433683989149905009273372586654059140405331146 33
K = 2.731599998459045235360287471352662497757247093699959574966967627 7
Integer{e*10^64}/10^64 = 2.71828182845904523536028747135266249775724709369995957496696762 77 = E

The recommended SI values of the physical constants C and K are now 299792458 and 2.7316, respectively. From a

physical point of view, C is the upper limit of the relative speed of circular harmonic motion, and K is the upper limit of the relative speed of radial harmonic motion. The nine-digit length of unit C limits the accuracy of rotation speed metric at the level of 10^(-8). The five-digit length of K limits the accuracy of the temperature metric at the level of 10^(-4). At the same time, modern quantum physics successively works with quantum units at the Boltzmann's level 10^(-23), the Dalton's level 10^(-29), the Planck's level 10^(-34) and Avogadro's level 10^(+23).

From a mathematical point of view, C and K are derived from the gradient of the perimeter of the circle relative to the radius gradient and vice versa. By defining the Planck unit as the relative inverse perimeter, the Boltzmann unit as the relative inverse radius, and the Dalton unit as the relative density of the velocity field, we can exclude the metric artifacts m, s, and kg as optional and, therefore, logically redundant units. This redefinition was recently made in the new SI-2019 metric system, but without changing the conditionally “exact” values of C and K. This is the main fallacy of modern metric - the use of very short nine-digit value for the unit of progressive speed, when the wave frequency can be many decimal orders greater. Yes, we can never completely eliminate the error of speed and frequency measurement, but we

always can calculate the boundaries of information entropy using “honest” computation arithmetic with floating, fixed and excluded points:

C(pi) = 299792457.867591338433683989149905009273372586654059140405331146 33
PI = 3.14159265358979323846264338327950288419716939937510582097494459 23
PI(C) = 3.14159265358979323846264338327950288419716939937510582097156248 30
[PI - PI(C)] = 0.00000000000000000000000000000000000000000000000000000000033821093

The mutual entropy between the units C and PI has a decimal order of 10^(-57), and this value can be considered as the entropy horizon for quantum calculus with a finite order of (-57) = (-23) + (-34), which is the sum of decimal orders of Boltzmann and Planck constants; and, therefore, the energy entropy of a quantum metric can be regarded as the equivalent of the informational entropy of a quantum calculus.

Fundamental theorem of calculus relates the derivative to the integral and provides the principal method for evaluating definite integrals. Any function that is continuous over an interval has an anti-derivative (a function whose rate of change equals the function) on that interval. The definite integral of such a function over an interval a < x < b is the difference F(b) - F(a), where F is the anti-derivative of the function. This theorem shows the inverse function relationship of the derivative and the integral and serves as the backbone of the physical sciences (britannica.com). The main logical fallacy of standard functional analysis is the conclusion that the numbers pi and e are mathematical constants. Sorry, but only integer numbers can be “honestly” called constants, because transcendental and irrational numbers in any digital notation are, in fact, variables, infinite sequences, in other words, progressive and backward waves of integers and of their inverse values. The corresponding “fair” mathematics of harmonic wave motion should be the dynamic mathematics.

We define the operator NM (natural median) as the arithmetic mean of four average values - the root mean square MR = Sqrt((pi^2+e^2)/2), the arithmetic mean MA = (pi+e)/2, the geometric mean MG = Sqrt(pi*e) and harmonic mean MH = 2/(1/pi+1/e). Med = Sum {Mi}/4. Obviously, MR > MA > MG > MH. But this is not always the case for numbers in a decimal normalized notation with the integer part and mantissa. For example, if we use symbolic and digital notation for the inverse power of pi and e, the results are catastrophically different (by 48 decimal orders).

1/pi^33 = Zero = 1/e^38, but 1/pi^32 = 1/10^16 = 1/e^37 and 1/(3.14159...)^129 = 1/10^64 = 1/(2.71828...)^148.

We cannot completely write down any irrational or transcendental number due to their infinite length. But we can estimate the final entropy of functional analysis using correctly truncated numbers pi and e. It will be shown that the informational entropy of direct and inverse functional analysis using floating point arithmetic cannot be better than 10^(-12...-22). In any case, the computational problem can be solved only with the help of “honest” arithmetic, excluding informational entropy caused by algorithmic approximations when extracting roots and calculating logarithms, hyperbolic and trigonometric functions.

Inverse exponential (or natural logarithmic) entropy is the main cause of the uncertainty in quantum physics and this can be clearly illustrated by the gauge X-factor from the equation of the relative wavelength-frequency displacement for a black body irradiation [Xi] = Root{X*e^X/(e^X - 1) = 5} = 4.9651142317442762999... or, with excluded point, in dependence of digital length:

X03 = 496
X06 = 496 511 - it is lower X-prime number
X09 = 496 511 423
X12 = 496 511 423 174
X15 = 496 511 423 174 427
X18 = 496 511 423 174 427 629
X21 = 496 511 423 174 427 629 999
X24 = 496 511 423 174 427 629 999 999
X27 = 496 511 423 174 427 629 999 999 999
X30 = 496 511 423 174 427 629 999 999 999 999
X33 = 496 511 423 174 427 629 999 999 999 999 999
X36 = 496 511 423 174 427 629 999 999 999 999 999 999
X39 = 496 511 423 174 427 629 999 999 999 999 999 999 999
X42 = 496 511 423 174 427 629 999 999 999 999 999 999 999 999
X45 = 496 511 423 174 427 629 999 999 999 999 999 999 999 999 999
X48 = 496 511 423 174 427 629 999 999 999 999 999 999 999 999 999 999
X49 = 496 511 423 174 427 629 999 999 999 999 999 999 999 999 999 999 9 - it is upper X-prime number .
X51 = 496 511 423 174 427 629 999 999 999 999 999 999 999 999 999 999 999
X54 = 496 511 423 174 427 629 999 999 999 999 999 999 999 999 999 999 999 999

X57 = 496 511 423 174 427 629 999 999 999 999 999 999 999 999 999 999 999 999 999
X60 = 496 511 423 174 427 629 999 999 999 999 999 999 999 999 999 999 999 999 999 999
X63 = 496 511 423 174 427 629 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999
X64 = 496 511 423 174 427 629 999 999 999 999 999 999 999 999 999 999 999 999 999 999 9

It can be shown that this exponential X-factor completely coordinates all fundamental quantum units of the standard physical model within the framework of the universal metric of relative space-time and combines the special theory of relativity with the general, associates the discrete and continuous mathematics, combines the standard and non-standard functional analysis with an accuracy 10^(-57), if, of course, we use “honest” arithmetic during operations with very small or very large quantum units. Let's continue estimation of computational entropy of standard functional analysis using arithmetic with floating point.

Squared sum of four means in symbolic notation:

[SMS] = (Sqrt((pi^2+e^2)/+(pi+e)/2+Sqrt(pi*e)+2/(1/pi+1/e))^2 = 136.9938985020083597 - 19 digit
Inverse value 1/[SMS] = 0.0072995951712794 - 17 digit

Natural median in decimal normalized notation:

[NM] = Med{e...pi} = Med{pi...e} = ((Sqrt[SMS]/4)*4)^2 = 2.9261098161852235 - 16 digit
[SMS] = (4*[NM])^2 = 136.9938985020083593447 - 22 digit

Here we got the natural boundaries of the computational entropy of [NM] and [SMS] which are functions of two transfinite variables (the infinitely growing numbers pi and e). Let's continue:

Ln(Sqrt(10))+59*Ln(10) = 136.9972503724980956 - 19 digit
1/(Ln(Sqrt(10))+59*Ln(10) = 0.0072990669227437 - 17 digit

Lim[1/Sum{729927/10^(8*N)}] = 137 = A
1/A = 0.0072992700729927 - 17 digit

Ln(pi)+59*Ln(10) = 137.0038130331457184 - 19 digit
1/(Ln(pi)+59*Ln(10)) = 0.0072994165742815 - 17 digit

Prime number A = 137 is the first center of infinite periodic mirror symmetry (super symmetry) of the inverse natural set. Anytime and anywhere (10^N)/A = <729927007299270072992700729270072992700729927>. This conclusion can be approved only by the full set of calculations from N = 1 to N = 64. Full set of calculations Med{e...pi} relatively module of complex number {137+j*5/10^64} is sufficient to define completely the alphabet of quantum calculus and corresponding geometry of subatomic space. The most outstanding and mysterious fact for many people is that always [abcdabcd]/[abcd] = 73*137, where a, b, c, d can be any different or any similar decimal digits from 0 to 9, for example, such as Albert Einstein's birthday in two different records 03140314/0314 = 14031403/1403 = 73*137 or my birthday 03050305/0305 = 73*137 = 05030503/0503 or your own birthday, my dear reader. You can take also the year instead a birthday with the same success.

My own arithmetical discoveries seem less mysterious:

C = (R/10^8+4*pi*C/10^18)^64*10^7=
=299792457.867591338433683989149905009273372586654059140405331146 33,
R = Integer{10^8*(C/10^7)^(1/64)} = 105456978. You can take as a first approximation to C any decimal digit from 0 to 9, and the result will be the same. From a stereo-metric point of view, C is a natural gauge unit of mutual coordination of solid angles in degree and in rad notations, since 4*pi rad = 720 deg.

Numbers AS and BS = 0.00602816999999> = 0.00602817 mutually connect binary (0...1), all decimal (0...10...100...1000...) and all qubit (000000>...111111>) calculation systems, because we always can write 0.99999> = 1, 9.9999> = 10 = 9*1.1111111>.

AS = 0.00729 = Lim{1/100/(1.11111111111111111>)^3}
AS = 0.00729 = 1/100/[10/(10-1)]^3
AS = 0.00729 =
=Lim{1/[A+(A+37*1)/10^3+{A+37*2)/10^6+(A+37*3)/10^9+(A+37*4)/10^12+(A+37*5)/10^15+...]}
AS = 0.00729 = (3*3*3*3*3*3)/[(2*2*2*2*2*2)*(5*5*5*5*5)]

AS = 0.00729 = 3^6/10^5.

Finite numbers AS = 0.00729 and BS = 0.00602817 = Sum{B/10^(3*N+8)} can be regarded as quantum super strings, obtained from infinite sets of digits 1 and 9.

+ 602 214 183 000 000 000 000 000 000
+ 000 602 214 183 000 000 000 000 000
+ 000 000 602 214 183 000 000 000 000
+ 000 000 000 602 214 183 000 000 000
+ 000 000 000 000 602 214 183 000 000
+ 000 000 000 000 000 602 214 183 000
+ 000 000 000 000 000 000 602 214 183

Sum = 602 816 999 999 999 999 999 397 183 and, of course, 602817 + 397183 = 1000000 = 10^6

Computational accuracy 10^(-15) seems quite sufficient for mono-molecular and mono-atomic quantum calculus since the experimentally estimated radius of atomic nuclear have such decimal order. The situation is much worse when calculating the square roots, harmonic and hyperbolic functions:

Arithmetical notation of angle 30 grad Sqrt(3)/2 = 0.8660254037844386 - accuracy 10^(-16)
Trigonometrical notation of angle 30 grad Cos(pi/6) = 0.866025403784 - accuracy 10^(-12)

Arithmetical "golden ratio" (Sqrt(5)+1)/2 = 1.6180339887498948 - accuracy 10^(-17)
Trigonometrical "golden ratio* 2*Cos(pi/5) = 1.61803398875 - accuracy 10^(-12)

Hyperbolic functions Cosh(19) - Sinh(19) = 89241150.48159364 - 89241150.48159364 = 0 - accuracy 10^(-16)
Cosh(18) - Sinh(18) = 32829984.568665262 - 32829984.568665247 = 0 + 15/10^9 = 0 - 85/10^9.

IV. Computation entropy of standard differential calculus

Entropy by definition is a measure of uncertainty of anything: of measurements, of calculations, of information processing. Well, there are no computer that could put in it's finite memory the exact values of infinite numbers. Let's determine the boundaries of informational entropy of the standard differential calculus using the numbers pi and e truncated to the decimal horizon 10^64, as for the constant C.

PI = (pi*10^64)/10^64 = 3.1415926535897932384626433832795028841971693993751058209749445923
E = (e*10^64)/10^64 = 2.7182818284590452353602874713526624977572470936999595749669676277

NDL = [PI - E] * [PI / E] * [E / PI] - full natural differential N - decimal order

0.42331082513074800310235591192684038643992230567514624600797696 45	64	
0.42331082513074800310235591192684038643992230567514624600797696 49	63	
0.42331082513074800310235591192684038643992230567514624600797696 99	62	absolute maximum
0.42331082513074800310235591192684038643992230567514624600797689 99	61	
0.42331082513074800310235591192684038643992230567514624600797699 99	60	
0.42331082513074800310235591192684038643992230567514624600798000 00	59	
0.42331082513074800310235591192684038643992230567514624600799999 99	58	same as 56, 57
0.42331082513074800310235591192684038643992230567514624600799999 99	57	same as 58, 56
0.42331082513074800310235591192684038643992230567514624600799999 99	56	same as 58, 57
0.42331082513074800310235591192684038643992230567514624600999999 99	55	
0.42331082513074800310235591192684038643992230567514624599999999 99	54	same as 53
0.42331082513074800310235591192684038643992230567514624599999999 99	53	same as 54
0.42331082513074800310235591192684038643992230567514625000000000 00	52	
0.42331082513074800310235591192684038643992230567514629999999999 99	51	
0.42331082513074800310235591192684038643992230567514599999999999 99	50	
0.42331082513074800310235591192684038643992230567514999999999999 99	49	
0.42331082513074800310235591192684038643992230567519999999999999 99	48	
0.42331082513074800310235591192684038643992230567599999999999999 99	47	
0.42331082513074800310235591192684038643992230567999999999999999 99	46	
0.42331082513074800310235591192684038643992230569999999999999999 99	45	

0.42331082513074800310235591192684038643992230599999999999999999999	44	
0.42331082513074800310235591192684038643992229999999999999999999999	43	same as 42
0.42331082513074800310235591192684038643992229999999999999999999999	42	same as 43
0.42331082513074800310235591192684038643992199999999999999999999999	41	
0.42331082513074800310235591192684038643991999999999999999999999999	40	
0.42331082513074800310235591192684038643989999999999999999999999999	39	
0.42331082513074800310235591192684038643999999999999999999999999999	38	same as 37
0.42331082513074800310235591192684038643999999999999999999999999999	37	same as 38
0.42331082513074800310235591192684038639999999999999999999999999999	36	
0.42331082513074800310235591192684038699999999999999999999999999999	35	
0.42331082513074800310235591192684038999999999999999999999999999999	34	
0.42331082513074800310235591192684039999999999999999999999999999999	33	
0.42331082513074800310235591192683999999999999999999999999999999999	32	same as 31
0.42331082513074800310235591192683999999999999999999999999999999999	31	same as 32
0.42331082513074800310235591192689999999999999999999999999999999999	30	
0.42331082513074800310235591192699999999999999999999999999999999999	29	
0.42331082513074800310235591189999999999999999999999999999999999999	28	
0.42331082513074800310235591199999999999999999999999999999999999999	27	
0.42331082513074800310235590999999999999999999999999999999999999999	26	
0.42331082513074800310235589999999999999999999999999999999999999999	25	
0.42331082513074800310235599999999999999999999999999999999999999999	24	
0.42331082513074800310235999999999999999999999999999999999999999999	23	
0.42331082513074800310239999999999999999999999999999999999999999999	22	
0.42331082513074800309999999999999999999999999999999999999999999999	21	same as 20
0.42331082513074800309999999999999999999999999999999999999999999999	20	same as 21
0.42331082513074800299999999999999999999999999999999999999999999999	19	
0.42331082513074799999999999999999999999999999999999999999999999999	18	same as 17, 16
0.42331082513074799999999999999999999999999999999999999999999999999	17	same as 18, 16
0.42331082513074799999999999999999999999999999999999999999999999999	16	same as 17, 18
0.42331082513074999999999999999999999999999999999999999999999999999	15	
0.42331082513069999999999999999999999999999999999999999999999999999	14	
0.42331082512999999999999999999999999999999999999999999999999999999	13	same as 12
0.42331082512999999999999999999999999999999999999999999999999999999	12	same as 13
0.42331082509999999999999999999999999999999999999999999999999999999	11	
0.42331082499999999999999999999999999999999999999999999999999999999	10	
0.42331082999999999999999999999999999999999999999999999999999999999	09	
0.42331079999999999999999999999999999999999999999999999999999999999	08	local minimum
0.42331099999999999999999999999999999999999999999999999999999999999	07	
0.42331000000000000000000000000000000000000000000000000000000000000	06	
0.42329999999999999999999999999999999999999999999999999999999999999	05	
0.42299999999999999999999999999999999999999999999999999999999999999	04	
0.42999999999999999999999999999999999999999999999999999999999999999	03	
0.40000000000000000000000000000000000000000000000000000000000000000	02	
1.00000000000000000000000000000000000000000000000000000000000000000	01	absolute minimum

The computational entropy of standard analysis has a “stairs” of uncertainty in decimal orders 12...13; 16...17...18; 20...21; 31...32; 37...38; 42...43, 53...54 and at the very end 56...57...58. But here we used the bracket record of the natural differential [ND] = [PI-E]*[PI/E]*[E/PI]. Let's try without the last four brackets. Any “smart” computing device instantly gives us the result [ND] = [PI-E], but the digital picture will change from a square matrix to a triangle matrix of rational numbers depending upon the digital length of PI and E.

NDE = [PI - E] - absolute natural difference	N - decimal order	
0.42331082513074800310235591192684038643992230567514624600797696 46	65	
0.42331082513074800310235591192684038643992230567514624600797696 5	64	
0.42331082513074800310235591192684038643992230567514624600797697	63	local maximum
0.42331082513074800310235591192684038643992230567514624600797696	62	local minimum
0.42331082513074800310235591192684038643992230567514624600797 7	61	
0.42331082513074800310235591192684038643992230567514624600798	60	
0.42331082513074800310235591192684038643992230567514624600800	59	same as 58

0.42331082513074800310235591192684038643992230567514624601	57 local maximum
0.42331082513074800310235591192684038643992230567514624601	56 local minimum same as 55
0.42331082513074800310235591192684038643992230567514624601	55 local minimum same as 56
0.42331082513074800310235591192684038643992230567514625	54
0.42331082513074800310235591192684038643992230567514625	53
0.42331082513074800310235591192684038643992230567514625	52
0.42331082513074800310235591192684038643992230567515	51
0.42331082513074800310235591192684038643992230567515	50
0.42331082513074800310235591192684038643992230567515	49
0.42331082513074800310235591192684038643992230568	48
0.42331082513074800310235591192684038643992230568	47
0.42331082513074800310235591192684038643992230568	46
0.42331082513074800310235591192684038643992230568	45 same as 44
0.42331082513074800310235591192684038643992230568	44 same as 45
0.42331082513074800310235591192684038643992230568	43
0.42331082513074800310235591192684038643992230568	42
0.42331082513074800310235591192684038643992230568	41
0.42331082513074800310235591192684038643992230568	40 local maximum same as 39
0.42331082513074800310235591192684038643992230568	39 local maximum same as 40
0.42331082513074800310235591192684038643992230568	38
0.42331082513074800310235591192684038643992230568	37
0.42331082513074800310235591192684038643992230568	36
0.42331082513074800310235591192684038643992230568	35
0.42331082513074800310235591192684038643992230568	34 local minimum same as 33
0.42331082513074800310235591192684038643992230568	33 local minimum same as 34
0.42331082513074800310235591192684038643992230568	32
0.42331082513074800310235591192684038643992230568	31
0.42331082513074800310235591192684038643992230568	30
0.42331082513074800310235591192684038643992230568	29
0.42331082513074800310235591192684038643992230568	28
0.42331082513074800310235591192684038643992230568	27
0.42331082513074800310235591192684038643992230568	26
0.42331082513074800310235591192684038643992230568	25
0.42331082513074800310235591192684038643992230568	24
0.42331082513074800310235591192684038643992230568	23
0.42331082513074800310235591192684038643992230568	22
0.42331082513074800310235591192684038643992230568	21 same as 20
0.42331082513074800310235591192684038643992230568	20 same as 21
0.42331082513074800310235591192684038643992230568	19
0.42331082513074800310235591192684038643992230568	18 local minimum same as 17, 16
0.42331082513074800310235591192684038643992230568	17 local minimum same as 18, 16
0.42331082513074800310235591192684038643992230568	16 local minimum same as 17, 18
0.42331082513074800310235591192684038643992230568	15
0.42331082513074800310235591192684038643992230568	14
0.42331082513074800310235591192684038643992230568	13 same as 12
0.42331082513074800310235591192684038643992230568	12 same as 13
0.42331082513074800310235591192684038643992230568	11
0.42331082513074800310235591192684038643992230568	10
0.42331082513074800310235591192684038643992230568	09
0.42331082513074800310235591192684038643992230568	08
0.42331082513074800310235591192684038643992230568	07 local maximum
0.42331082513074800310235591192684038643992230568	06
0.42331082513074800310235591192684038643992230568	05
0.42331082513074800310235591192684038643992230568	04
0.42331082513074800310235591192684038643992230568	03
0.42331082513074800310235591192684038643992230568	02
1.	01 absolute minimum

Comparing with square matrix the shift of entropy "stairs" is not more than 1 decimal order.

V. Computational entropy of non-standard differential calculus

Non-standard functional analysis can be defined as integral-differential calculus in the decimal normalized notation [CN...K] where CN = C/10^8

QDL = [CN - K] * [CN / K] * [K / CN] - full speed differential

	N - decimal order	
0.266324580216868148976552420146387594976478772840631829086343835 6	64	
0.266324580216868148976552420146387594976478772840631829086343835 8	63	
0.266324580216868148976552420146387594976478772840631829086343839 9	62	absolute maximum
0.266324580216868148976552420146387594976478772840631829086343799 9	61	local minimum
0.266324580216868148976552420146387594976478772840631829086343999 9	60	
0.266324580216868148976552420146387594976478772840631829086349999 9	59	
0.266324580216868148976552420146387594976478772840631829086399999 9	58	
0.266324580216868148976552420146387594976478772840631829086999999 9	57	
0.266324580216868148976552420146387594976478772840631829089999999 9	56	
0.266324580216868148976552420146387594976478772840631829099999999 9	55	
0.266324580216868148976552420146387594976478772840631829999999999 9	54	same as 53
0.266324580216868148976552420146387594976478772840631829999999999 9	53	same as 54
0.266324580216868148976552420146387594976478772840631899999999999 9	52	
0.266324580216868148976552420146387594976478772840631999999999999 9	51	
0.266324580216868148976552420146387594976478772840639999999999999 9	50	
0.266324580216868148976552420146387594976478772840599999999999999 9	49	local minimum
0.266324580216868148976552420146387594976478772840999999999999999 9	48	
0.266324580216868148976552420146387594976478772849999999999999999 9	47	
0.266324580216868148976552420146387594976478772899999999999999999 9	46	
0.266324580216868148976552420146387594976478772999999999999999999 9	45	
0.266324580216868148976552420146387594976478769999999999999999999 9	44	local minimum
0.266324580216868148976552420146387594976478799999999999999999999 9	43	
0.266324580216868148976552420146387594976477999999999999999999999 9	42	local minimum
0.266324580216868148976552420146387594976479999999999999999999999 9	41	
0.266324580216868148976552420146387594976499999999999999999999999 9	40	
0.266324580216868148976552420146387594975999999999999999999999999 9	39	local minimum
0.266324580216868148976552420146387594979999999999999999999999999 9	38	
0.266324580216868148976552420146387594999999999999999999999999999 9	37	same as 36
0.266324580216868148976552420146387594999999999999999999999999999 9	36	same as 37
0.266324580216868148976552420146387599999999999999999999999999999 9	35	same as 34
0.266324580216868148976552420146387599999999999999999999999999999 9	34	same as 35
0.266324580216868148976552420146387999999999999999999999999999999 9	33	
0.266324580216868148976552420146389999999999999999999999999999999 9	32	
0.266324580216868148976552420146399999999999999999999999999999999 9	31	
0.266324580216868148976552420147000000000000000000000000000000000 0	30	local maximum
0.266324580216868148976552420139999999999999999999999999999999999 9	29	
0.266324580216868148976552420099999999999999999999999999999999999 9	28	
0.266324580216868148976552419999999999999999999999999999999999999 9	27	same as 26
0.266324580216868148976552419999999999999999999999999999999999999 9	26	same as 27
0.266324580216868148976552399999999999999999999999999999999999999 9	25	
0.266324580216868148976551999999999999999999999999999999999999999 9	24	
0.266324580216868148976549999999999999999999999999999999999999999 9	23	
0.266324580216868148976599999999999999999999999999999999999999999 9	22	
0.266324580216868148975999999999999999999999999999999999999999999 9	21	
0.266324580216868148969999999999999999999999999999999999999999999 9	20	local minimum
0.266324580216868148999999999999999999999999999999999999999999999 9	19	same as 18
0.266324580216868148999999999999999999999999999999999999999999999 9	18	same as 19
0.266324580216868150000000000000000000000000000000000000000000000 0	17	
0.266324580216868099999999999999999999999999999999999999999999999 9	16	
0.266324580216867999999999999999999999999999999999999999999999999 9	15	local minimum
0.266324580216869999999999999999999999999999999999999999999999999 9	14	
0.266324580216899999999999999999999999999999999999999999999999999 9	13	
0.266324580215999999999999999999999999999999999999999999999999999 9	12	local minimum
0.266324580219999999999999999999999999999999999999999999999999999 9	11	

0.266324580199999999999999999999999999999999999999999999999999	10	
0.266324579999999999999999999999999999999999999999999999999999	09	local minimum same as 08
0.266324579999999999999999999999999999999999999999999999999999	08	local minimum same as 09
0.266324599999999999999999999999999999999999999999999999999999	07	
0.266324999999999999999999999999999999999999999999999999999999	06	
0.266329999999999999999999999999999999999999999999999999999999	05	
0.266399999999999999999999999999999999999999999999999999999999	04	
0.265999999999999999999999999999999999999999999999999999999999	03	
0.259999999999999999999999999999999999999999999999999999999999	02	
0.200000000000000000000000000000000000000000000000000000000000	01	
0.000000000000000000000000000000000000000000000000000000000000	00	absolute minimum

QDE = [CN - K] - absolute speed difference N - decimal order

0.266324580216868148976552420146387594976478772840631829086344	64	
0.266324580216868148976552420146387594976478772840631829086344	63	absolute maximum
0.266324580216868148976552420146387594976478772840631829086344	62	
0.266324580216868148976552420146387594976478772840631829086344	61	local minimum
0.266324580216868148976552420146387594976478772840631829086344	60	
0.266324580216868148976552420146387594976478772840631829086344	59	
0.266324580216868148976552420146387594976478772840631829086344	58	
0.266324580216868148976552420146387594976478772840631829086344	57	
0.266324580216868148976552420146387594976478772840631829086344	56	
0.266324580216868148976552420146387594976478772840631829086344	55	
0.266324580216868148976552420146387594976478772840631829086344	54	same as 53
0.266324580216868148976552420146387594976478772840631829086344	53	same as 54
0.266324580216868148976552420146387594976478772840631829086344	52	
0.266324580216868148976552420146387594976478772840631829086344	51	
0.266324580216868148976552420146387594976478772840631829086344	50	
0.266324580216868148976552420146387594976478772840631829086344	49	local minimum
0.266324580216868148976552420146387594976478772840631829086344	48	
0.266324580216868148976552420146387594976478772840631829086344	47	
0.266324580216868148976552420146387594976478772840631829086344	46	
0.266324580216868148976552420146387594976478772840631829086344	45	
0.266324580216868148976552420146387594976478772840631829086344	44	local minimum
0.266324580216868148976552420146387594976478772840631829086344	43	
0.266324580216868148976552420146387594976478772840631829086344	42	local minimum
0.266324580216868148976552420146387594976478772840631829086344	41	
0.266324580216868148976552420146387594976478772840631829086344	40	local maximum
0.266324580216868148976552420146387594976478772840631829086344	39	local minimum
0.266324580216868148976552420146387594976478772840631829086344	38	
0.266324580216868148976552420146387594976478772840631829086344	37	same as 36
0.266324580216868148976552420146387594976478772840631829086344	36	same as 37
0.266324580216868148976552420146387594976478772840631829086344	35	same as 34
0.266324580216868148976552420146387594976478772840631829086344	34	same as 35
0.266324580216868148976552420146387594976478772840631829086344	33	
0.266324580216868148976552420146387594976478772840631829086344	32	
0.266324580216868148976552420146387594976478772840631829086344	31	
0.266324580216868148976552420146387594976478772840631829086344	30	local maximum
0.266324580216868148976552420146387594976478772840631829086344	29	
0.266324580216868148976552420146387594976478772840631829086344	28	
0.266324580216868148976552420146387594976478772840631829086344	27	same as 26
0.266324580216868148976552420146387594976478772840631829086344	26	same as 27
0.266324580216868148976552420146387594976478772840631829086344	25	
0.266324580216868148976552420146387594976478772840631829086344	24	
0.266324580216868148976552420146387594976478772840631829086344	23	
0.266324580216868148976552420146387594976478772840631829086344	22	
0.266324580216868148976552420146387594976478772840631829086344	21	
0.266324580216868148976552420146387594976478772840631829086344	20	local minimum
0.266324580216868148976552420146387594976478772840631829086344	19	same as 18

0.266324580216868149	18	same as 19
0.26632458021686815	17	local maximum
0.2663245802168681	16	
0.266324580216868	15	local minimum
0.26632458021687	14	
0.2663245802169	13	
0.266324580216	12	local minimum
0.26632458022	11	
0.2663245802	10	
0.266324580	09	local minimum same as 08
0.26632458	08	local minimum same as 09
0.2663246	07	
0.266325	06	
0.26633	05	
0.2664	04	
0.266	03	
0.26	02	
0.2	01	
0.	00	absolute minimum

VI. Computation entropy of [C...T] calculus

C = 299792457.867591338433683989149905009273372586654059140405331146330000000
T = 299792456.086181602910895246782555676989610054855827164649691241464767561

CTD = [C - T] * [C / T] * [T / C] - full speed-temperature differential N - decimal order

0.000000017814097355227887423673493322837625317982319757556320221835232438	64	
0.000000017814097355227887423673493322837625317982319757556320221535232440	63	
0.000000017814097355227887423673493322837625317982319757556320218535232499	62	
0.000000017814097355227887423673493322837625317982319757556320158535232999	61	
0.000000017814097355227887423673493322837625317982319757556319758535239997	60	
0.000000017814097355227887423673493322837625317982319757556318758535299999	59	
0.000000017814097355227887423673493322837625317982319757556308758535999999	58	
0.000000017814097355227887423673493322837625317982319757556008758540000000	57	
0.000000017814097355227887423673493322837625317982319757553008758599999998	56	
0.000000017814097355227887423673493322837625317982319757503008758999999999	55	
0.000000017814097355227887423673493322837625317982319757503008759999999997	54	
0.000000017814097355227887423673493322837625317982319753503008800000000001	53	
0.000000017814097355227887423673493322837625317982319753503008999999999998	52	
0.000000017814097355227887423673493322837625317982319353503009999999999999	51	
0.000000017814097355227887423673493322837625317982318353503099999999999999	50	
0.000000017814097355227887423673493322837625317982228353503999999999999998	49	
0.000000017814097355227887423673493322837625317981728353509999999999999998	48	
0.000000017814097355227887423673493322837625317981728353599999999999999999	47	
0.000000017814097355227887423673493322837625317941728353999999999999999999	46	
0.000000017814097355227887423673493322837625317441728359999999999999999998	45	
0.000000017814097355227887423673493322837625311441728399999999999999999998	44	
0.000000017814097355227887423673493322837625251441728999999999999999999999	43	
0.000000017814097355227887423673493322837624451441730000000000000000000001	42	
0.000000017814097355227887423673493322837619451441800000000000000000000000	41	
0.000000017814097355227887423673493322837599451441999999999999999999999999	40	
0.000000017814097355227887423673493322836899451449999999999999999999999999	39	
0.000000017814097355227887423673493322833899451499999999999999999999999997	38	
0.000000017814097355227887423673493322803899452000000000000000000000000000	37	
0.000000017814097355227887423673493322103899459999999999999999999999999999	36	
0.000000017814097355227887423673493320103899499999999999999999999999999998	35	
0.000000017814097355227887423673493230103899999999999999999999999999999999	34	same as 33, 32
0.000000017814097355227887423673493230103899999999999999999999999999999999	33	same as 34, 32
0.000000017814097355227887423673493230103899999999999999999999999999999999	32	same as 33, 34

0.00000001781409735522788742367344323010399999999999999999999999999999999 31
0.00000001781409735522788742367344323011000000000000000000000000000000000 30
0.00000001781409735522788742366444323019999999999999999999999999999999998 29
0.00000001781409735522788742357444323099999999999999999999999999999999998 28
0.00000001781409735522788742317444323999999999999999999999999999999999999 27
0.00000001781409735522788742217444329999999999999999999999999999999999999 26
0.00000001781409735522788733217444399999999999999999999999999999999999998 25
0.00000001781409735522788653217445000000000000000000000000000000000000000 24
0.00000001781409735522787753217449999999999999999999999999999999999999999 23
0.00000001781409735522784753217499999999999999999999999999999999999999998 22
0.00000001781409735522704753218000000000000000000000000000000000000000000 21
0.00000001781409735522104753219999999999999999999999999999999999999999999 20
0.00000001781409735519104753299999999999999999999999999999999999999999999 19
0.00000001781409735489104754000000000000000000000000000000000000000000000 18
0.00000001781409735089104759999999999999999999999999999999999999999999999 17
0.00000001781409727089104799999999999999999999999999999999999999999999997 16
0.00000001781409697089104999999999999999999999999999999999999999999999998 15
0.00000001781409397089110000000000000000000000000000000000000000000000000 14
0.00000001781408397089199999999999999999999999999999999999999999999999999 13
0.00000001781318397089999999999999999999999999999999999999999999999999999 12
0.00000001780818397090000000000000000000000000000000000000000000000000000 11
0.00000001773818397099999999999999999999999999999999999999999999999999998 10
0.00000001713818397999999999999999999999999999999999999999999999999999998 09
0.00000000913818399999999999999999999999999999999999999999999999999999998 08

Relative speed-temperature differential analysis has only one "stair" of entropy at the decimal orders 32...33...34. This corresponds to Planck's energy quantum and is 20 orders better than entropy of standard differential analysis.

VII. Entropy of microwave background temperature

E+AS+BS = 2.731599998459045235360287471352662497757247093699959574966967627 7 = K
2.728587578186839052494490511181326398125596799716452122330953527 2
2.728585913459045235360287471352662497757247093699959574966967627 7
Med{TA...K} = 2.728585081094640496077789039799006640392282592068731127547728599 1 = TAK
2.728584248730235756485459637525478763377227150519528988426107714 6
2.728582584002441939970918539136558902309059324338983824466885527 0
EA+AS = 2.725571828459045235360287471352662497757247093699959574966967627 7 = TA
2.721926828459045235360287471352662497757247093699959574966967627 7
2.721925608180930353875436113969506365967493203491907048301857391 6
Med{TA...E} = 2.721925608180930353875436113969506365967493203491907048301857391 6 = TAE
2.721924387902815471409553189183560331361367797441434987166381082 4
2.721921947348773978887853484396129496888288517274048935176855960 2
2.718281828459045235360287471352662497757247093699959574966967627 7 = E

TAK = 2.728585081094640496077789039799006640392282592068731127547728599 1 - upper limit
2.725257378784926200287693149870647768904985748710153428447408810 4
2.725255344637785424976612576884256503179887897780319087924792995 4
Med{TAE...TAK} = 2.725254327563455887147733463147219551003800684099524782984184550 8 = TBG
2.725253310489126348752219626475580965159968208920225213364549506 7
2.725251276341985574574408499358392966770360880987401402199986890 8
TAE = 2.721925608180930353875436113969506365967493203491907048301857391 6 - lower limit

In cosmology, TBG is considered as the relic temperature of the universe, and its last measured value is about 2.7252 K.

The relic temperature is not included as a fundamental unit in quantum physics, and I believe that this is one of the main methodological errors of SI metric, since TBG is inherently an energy unit coordinating classic and quantum physics.

VIII. Entropy of progressive temperature

RE^64*10^7 = 299792456.257274188286886027303032761337552565628547217370703488399000000 0 = TE

299792456.167764350560368025292578407819031741202951788582991254525016000
299792456.167764350547005429124154112151435989941575148664845770865500000
Med{TA...TE} = 299792456.167764350540324131039941964317340309992120939261859473997920567 6 = TAE
299792456.167764350533642832955729816483244630042666729857306682079436270 3
299792456.167764350520280236787305520815648878781290089942294188521730000 0
RA^64*10^7 = 299792456.078254512807124830945275462965319414254603079958988053332000000 0 = TA
299792456.004598855306139229097696478566058727061813308924036162544346000 0
299792456.004598855297091042969970545806618232826695081236430750132500000 0
Med{TA...TK} = 299792456.004598855292566949906107579426761441792566038565347010739400031 9 = TAK
299792456.004598855288042856842244613046904650758436995892933555723628127 7
299792456.004598855278994670714518680287464156523318768207987574557126999 9
RK^64*10^7 = 299792455.930943197787057254994665628647917051398787082513873446933000000 0 = TK

TAE = 299792456.167764350540324131039941964317340309992120939261859473997920567 6
299792456.086181602927546127853962961636521490143627276813730583570776167 2
299792456.086181602916445540473024771872050875892343488913603242368660299 8
Pulsation speed T = 299792456.086181602910895246782555676989610054855827164649699124146476756 1
299792456.086181602905344953092086582107169233819310840385699042327460919 0
299792456.086181602894244365711148392342698619568027052485763628319009638 2
TAK = 299792456.004598855292566949906107579426761441792566038565347010739400031 9

Rotation speed C = 299792457.867591338433683989149905009273372586654059140405331146330000000 0
299792456.976886471995463605487553258872520648803815186609877322786110592 9
299792456.976886470672289617966230343131491320754943152527515135238238378 1
Translation speed V = 299792456.976886470010702624205568882340974234207813146779679430223758144 6
299792456.976886469349115630444907421550457147660683128145940543307168928 2
299792456.976886468025941642923584505809427819611811119835384719563514679 2
Pulsation speed T = 299792456.086181602910895246782555676989610054855827164649699124146476756 1

Constants T and V give us the possibility to coordinate mutually all fundamental constants of standard physical model at the decimal level 10^(-34), but for today they are not basic metric units, and this is one more logical fallacy of new SI.

Conclusion

The unified information-entropy-energy paradigm of quantum physics, quantum chemistry and quantum computing is logically developed and substantiated on the base. In general the results obtained can be regarded as Quantum Renaissance of Classic Physics.

References

[1]. Eugene Machusky. "Quantum Metric of Classic Physics", IOP Conference Series: Material Science and Engineering, Online ISSN: 1757-899X, vol. 239, pp. 48-54, 2017.

[2]. Eugene Machusky. "Complex Geometry of Wave Motion", International Journal of Engineering and Technology, vol. 10, no. 2, pp. 184-188, 2018.

[3]. Eugene Machusky. "Natural Qubit Matrix of Primary Elements of Matter", MATEC Web of Conferences 186, 01005, 2018.

Об одном подходе по выявлению кибератак на киберфизические системы с применением глубокой гибридной модели

Расим Алигулиев[1], Ядигар Имамвердиев[1], Рамиз Шыхалиев[1], Людмила Сухостат[1]

1 Институт информационных технологий, Национальная Академия Наук Азербайджана, АЗЕРБАЙДЖАН, г. Баку, ул. Б.Вахабзаде, 9А,
E-mail: r.alguliev@gmail.com, yadigarimam@gmail.com, shikhramiz61@gmail.com, lsuhostat@hotmail.com

The paper introduces a hybrid model, combining the advantages of deep neural networks for detecting cyber-attacks on cyber-physical system failure. The AdamW+Amsgrad optimizer is used to reduce the model's training error and increase the learning speed. Experiments show a significant increase in the accuracy of malfunction detection caused by cyber-attacks.

Ключевые слова – кибератака, киберфизическая система, глубокое обучение, обнаружение кибератак, оптимизатор Adam.

I. Введение

С точки зрения информационной безопасности, глобальная доступность объектов киберпространства породила проблему обеспечения устойчивой работы современного производства в контексте случайных и целевых компьютерных атак. Существующие угрозы, в первую очередь, направлены на установление контроля над актуаторами в промышленной, энергетической, социально-политической и других сферах.

Обнаружение аномального или аварийного состояния киберфизических систем (КФС) может быть реализовано с использованием многочисленных подходов [1]. Актуальность интеграции технологии машинного обучения с КФС подчеркивается большим количеством научных исследований по этой теме. Традиционные классификаторы машинного обучения, в том числе метод k-ближайших соседей, наивный Байес, деревья решений и машины опорных векторов, были использованы для обнаружения кибератак на КФС.

Однако эффективная диагностика кибератак на КФС остается трудной задачей, когда системы очень сложны, а знаний об исследуемом объекте недостаточно. Глубокие нейронные сети могут эффективно решить проблему адаптации к обнаружению кибератак на КФС [2].

Глубокое обучение – это современная методика искусственного интеллекта, состоящая из алгоритмов и решений, которые значительно расширяют возможности и эффективность нейронных сетей [3]. Большое количество слоев позволяет нейронной сети выстраивать концепцию исследуемого объекта из простых характеристик, постепенно переходя к более сложным.

В данной работе предлагается гибридная модель на основе глубоких нейронных сетей. Для повышения точности и стабильности предлагаемой модели в качестве оптимизатора рассматривается алгоритм AdamW+Amsgrad [4]. Оценка предложенной модели выполняется на наборе данных SWaT (Secure Water Treatment) [5], который содержит информацию о нормальном состоянии КФС и ее отказах, вызванных кибератаками.

Подводя итоги исследований по выявлению неисправностей КФС, вызванных кибератаками, можно сделать следующие выводы: во-первых, необходимо обеспечить информационную безопасность КФС приложений в связи с постоянным ростом объемов данных. Во-вторых, недостаточно исследований было посвящено использованию ансамблей глубоких нейронных сетей, которые повышают точность обнаружения атак на КФС. Кроме того, не учитываются все преимущества глубоких нейронных сетей. Это подтверждает актуальность нашего исследования.

II. Предлагаемый подход

На Рис.1 показана структура предлагаемой глубокой гибридной модели для обнаружения кибератак на КФС. Модель сочетает в себе преимущества глубокой управляемой рекуррентной нейронной сети (gated recurrent unit neural network, GRU), одномерной глубокой сверточной нейронной сети (convolutional neural network, CNN) и глубокой нейронной сети с долгой краткосрочной памятью (long short-term memory neural network, LSTM) для обнаружения кибератак, приводящих к отказам системы.

На этапе обучения применяется оптимизатор AdamW+Amsgrad. Правило обновления определяется следующим образом:

$$w_{i+1} = (1-\lambda)w_i - \frac{\alpha}{\sqrt{\hat{S}_t}+\varepsilon}V_t, \quad (1)$$

где

$$\hat{S}_t = \max(\hat{S}_{t-1}, S_t), \quad (2)$$

$$V_t = \beta_1 V_{t-1} + (1-\beta_1)g_t, \quad (3)$$

$$S_t = \beta_2 S_{t-1} + (1-\beta_2)g_t^2. \quad (4)$$

Здесь λ - сокращение весов, α - скорость обучения, V_t - момент 1-го порядка, S_t - момент 2-го порядка, β_1 - скорость затухания момента 1-го порядка, β_2 - скорость затухания момента 2-го порядка и g_t – градиент для шага t.

Предлагаемый подход обеспечивает высокую точность обнаружения кибератак на КФС по сравнению с известными методами машинного обучения.

III. Результаты экспериментов

Эксперименты проводились на Intel Xeon (R), CPU X5670 @ 2.93 ГГц×24, 24 Гб RAM. Подход

оценивался на языке Python 2.7.13 с использование различных библиотек, включая Tensorflow и Keras.

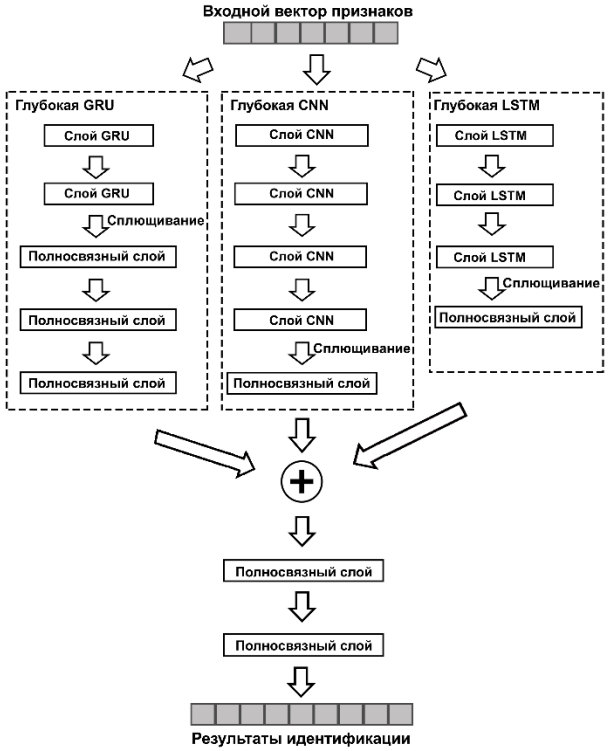


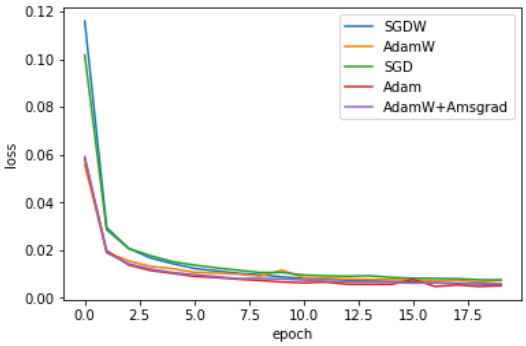
Рис.1. Структура предлагаемой гибридной модели

Экспериментальный набор данных SWaT был собран в Сингапурском университете технологий и дизайна [5] и содержит данные системы очистки сырой воды, которая обычно используется в общественной инфраструктуре. Он включает информацию о 36 атаках.

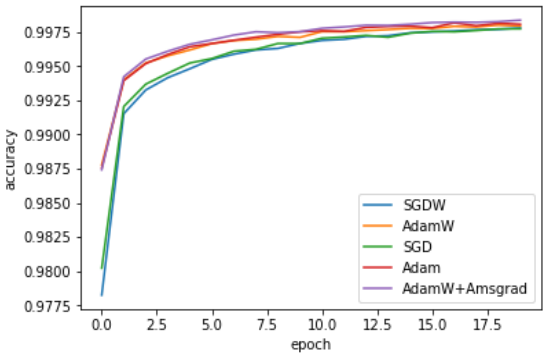
Предлагаемый подход оценивается с применением различных оптимизаторов: стохастического градиентного спуска (SGD), SGD с разобщенным сокращением веса (SGDW), оптимизатора Adam, Adam с разобщенным сокращением веса (AdamW) и AdamW+Amsgrad.

На Рис.2 приведены полученные результаты. Ошибка тестирования стабильно уменьшается с увеличением количества эпох. Однако при использовании оптимизатора SGDW предлагаемая модель показала более высокий уровень ошибок.

Наилучший результат показал предложенный подход с оптимизатором AdamW+Amsgrad, тогда как SGDW был менее точным.



а)



б)

Рис.2. Кривые потерь и точности обнаружения кибератак при различных оптимизаторах

По точности все оптимизаторы показали высокие результаты при обнаружении кибератак, достигнув 99,68%, 99,75%, 99,67%, 99,79% и 99,80% для SGDW, AdamW, SGD, Adam и AdamW+Amsgrad соответственно.

Также предложенная модель была оценена с помощью других методов машинного обучения (глубокой нейронной сети и машин опорных векторов) и показала более точный результат.

Выводы

Данная статья была направлена на обнаружение кибератак на КФС. Была продемонстрирована эффективность предложенной глубокой гибридной модели, объединяющей преимущества трех глубоких нейронных сетей. Модель тестировалась при различных оптимизаторах. Сравнение предложенной модели с применением AdamW+Amsgrad и известных методов машинного обучения показало более точный результат при рассмотрении предлагаемого подхода.

Литература

[1] Zeadally, S. Cyber-physical system design with sensor networking technologies / S. Zeadally, N. Jabeur. - London: The Institution of Engineering and Technology, 2016. – 368 p.

[2] Kravchik, M. Detecting cyber attacks in industrial control systems using convolutional neural networks / M. Kravchik, A. Shabtai // Proceedings of the 4th CPS-SPC Workshop on Cyber-Physical Systems Security and PrivaCy, October 2018. - Toronto, Canada, 2018. – pp. 72-83.

[3] LeCun, Y. Deep learning / Y. LeCun, Y. Bengio, G. Hinton // Nature. - 2015. –No 521. – C. 436–44.

[4] Loshchilov, I. Decoupled weight decay regularization / I. Loshchilov, F. Hutter // Proceedings of the International Conference on Learning Representations (ICLR), May 2019. – New Orleans, USA, 2019. – pp. 1-19.

[5] Набор данных SWaT [Электронный ресурс]. – Режим доступа:http://itrust.sutd.edu.sg/research/testbeds/secure-water-treatment-swat/. – 12.09.2019 г.

Using Event Management Systems to Block Attacks on Networks

Albarghathi Mohammed Adrees Abdullah,
Oleksandr Fediushyn

Department of Information Technologies Security, Kharkiv National University of Radio Electronics, UKRAINE, Kharkiv, pr.Nauki,14,
E-mail: mohamed.albarghati@gmail.com, oleksandr.fediushyn@nure.ua

Annotation – A comparative analysis of the characteristics of modern event management systems to block attacks on networks was performed. The comparison this system was performed on the following indicators: monitoring and audit level, response type, adaptive capacity, architecture, detection method supported for platform deployment, principle of construction, method of extending functionality.

Keywords – security information and event management, network security, log management, IDS.

I. Introduction

The Internet is a large network that connects people around the world. Companies have networks that connect their employees to each other, and some people have networks in their homes that connect them to family members.

But like all things, there are threats that can affect networks; threats that could potentially cause service interruption, or harm. Because of this, it is important to protect them against those types of eventualities. One solution is to implement network security, and to perform a network security audit on a regular basis.

Network security is the collection of hardware and software tools that protect a company's network infrastructure. They are intended to address a number of possible threats that include:

Unauthorized Access - any user trying to gain access without the proper credentials. For example, a malicious user logging in from the outside.

Malicious Use - any user trying to perform something they shouldn't. For example, a user trying to delete important information.

Faults - any piece of software or device that fails in some way. For example, a printer that runs out of toner.

Tampering - any action that changes a piece of software or a device such that it behaves differently than it should. For example, changing the configuration of a secured door so that it can be opened without a key or credentials.

Destruction - any fault that is created in a willful fashion. For example, breaking a mouse or keyboard.

Disclosure - revealing important information. For example, letting intellectual property fall into a competitor's hands.

Network security is achieved by various tools including firewalls and proxy servers, encryption, logical security and access controls, anti-virus software, and auditing systems such as log management.

II. Problem analysis

As more businesses operate online, it's increasingly important to incorporate cybersecurity tools and threat detection to prevent downtime. Unfortunately, many unscrupulous cyber attackers are active on the web, just waiting to strike vulnerable systems. Security Information and Event Management (SIEM) products have become a core part of identifying and addressing cyber attacks.

SIEM stands for Security Information and Event Management. A SIEM [2-4] system provides real-time analysis of security alerts generated by applications and network hardware.

This term is somewhat of an umbrella for security software packages ranging from Log Management Systems to Security Log / Event Management, Security Information Management, and Security Event correlation. While a SIEM system isn't foolproof, it's one of the key indicators that an organization has a clearly defined cybersecurity policy. Nine times out of ten, cyber attacks don't have any clear tells on a surface level. To detect threats, it's more effective to use the log files. The superior log management capabilities of SIEMs have made them a central hub of network transparency.

Most security programs operate on a micro scale, addressing smaller threats but missing the bigger picture of cyber threats. An Intrusion Detection System (IDS) alone can seldom do more than monitor packets and IP addresses. Likewise, your service logs only show user sessions and configuration changes. SIEM puts these systems and others like it together to provide a complete overview of any security incident through real-time monitoring and the analysis of event logs.

Security information management (SIM) is the collection, monitoring and analysis of security-related data from computer logs. Also referred to as log management.

Security event management (SEM) is the practice of network event management including real-time threat analysis, visualization and incident response.

III. Solve a problem

SIEM's basic capabilities are as follows:

- Log Collection;
- Normalization – collecting logs and normalizing them into a standard format).
- Notifications and Alerts – notifying the user when security threats are identified
- Security Incident Detection.

Threat response workflow – workflow for handling past security events SIEM records data from across a users' internal network of tools and identifies potential issues and attacks. The system operates under a statistical model to analyze log entries. SIEM distributes collection agents and recalls data from the network, devices, servers, and firewalls. All this information is then passed to a management console where it can be analyzed to address emerging threats. It's not uncommon for advanced SIEM systems to use automated responses, entity behavior analytics and security orchestration. This ensures that

vulnerabilities between cybersecurity tools can be monitored and addressed by SIEM technology.

Once the necessary information reaches the management console, it is then viewed by a data analyst who can provide feedback on the overall process. This is important because feedback helps to educate the SIEM system in terms of machine learning and increasing its familiarity with the surrounding environment.

Once the SIEM software system identifies a threat, it then communicates with other security systems on the device to stop the unwanted activity. The collaborative nature of SIEM systems makes them a popular enterprise-scale solution. However, the rise of pervasive cyber threats has made many small- and mid-sized businesses consider the merits of a SIEM system as well.

SIEM has become a core security component of modern organizations. The main reason is that every user or tracker leaves behind a virtual trail in a network's log data. SIEM systems are designed to use this log data in order to generate insight into past attacks and events. A SIEM system not only identifies that an attack has happened, but allows you to see how and why it happened as well.

As organizations update and upscale to increasingly complex IT infrastructures, SIEM has become even more important in recent years. Contrary to popular belief, firewalls and antivirus packages are not enough to protect a network in its entirety. Zero-day attacks can still penetrate a system's defenses even with these security measures in place.

SIEM addresses this problem by detecting attack activity and assessing it against past behavior on the network. A SIEM system has the ability to distinguish between legitimate use and a malicious attack. This helps to increase a system's incident protection and avoid damage to systems and virtual property.

The use of SIEM also helps companies to comply with a variety of industry cyber management regulations. Log management is the industry standard method of auditing activity on an IT network. SIEM systems provide the best way to meet this regulatory requirement and provide transparency over logs in order to generate clear insights and improvements.

1. SolarWinds Security Event Manager

One of the most competitive SIEM tools on the market with a wide range of log management features. The real-time incident response makes it easy to actively manage your infrastructure and the detailed and intuitive dashboard makes this one of the easiest to use on the market. With 24/7 support, this is a clear choice for SIEM.

2. ManageEngine EventLog Analyzer (FREE TRIAL) – A SIEM tool that manages, protects, and mines log files. This system installs on Windows, Windows Server, and Linux.

3. Micro Focus ArcSight ESM – Comprehensive SIEM tool that runs on Windows environments and is very well suited to large organizations.

4. Splunk Enterprise Security – This tool for Windows and Linux is a world leader because it combines network analysis with log management together with an excellent analysis tool.

5. LogRhythm Security Intelligence Platform – Cutting-edge AI-based technology underpins this traffic and log analysis tool for Windows and Linux.

6. AlienVault Unified Security Management – Great value SIEM that runs on Mac OS as well as Windows.

7. RSA NetWitness – Extremely comprehensive and tailored towards large organizations but a bit too much for small and medium-sized enterprises. Runs on Windows.

8. IBM Qradar – Market-leading SIEM tool that runs on Windows environments.

9. McAfee Enterprise Security Manager – Popular SIEM tool that runs through your Active Directory records to confirm system security. Runs on Mac OS as well as Windows.

Conclusions

Not all SIEM systems are built the same. As a result, there is no one-size-fits-all solution. A SIEM solution that's right for one company may be incomplete to another. As mentioned above, log data management is a core component of any enterprise-scale SIEM system. A SIEM system needs to pool log data from a variety of different sources, each with their own way of categorizing and recording data. When looking for a SIEM system, you want one that has the ability to normalize data effectively.

In terms of convenience and regulatory requirements, having a SIEM with extensive compliance reporting features is very important. In general, most SIEM systems have some kind of onboard report generating system that will help you to conform to your compliance requirements. If a breach or attack occurs, you can generate a report that details how it happened extensively. You can then use this data to refine internal processes and make adjustments to your network infrastructure to make sure it doesn't happen again. This uses SIEM technology keeps your network infrastructure evolving to address new threats.

Having the ability to set the criteria for future security alerts is essential for maintaining an effective SIEM system through threat intelligence. Refining alerts is the main way you keep your SIEM system updated against new threats. Innovative cyber attacks are emerging every day, so using a system that's designed to add new security alerts stops you from getting left behind.

References

[1] Nicolett.M., Williams.A.T., Proctor.P.E. 'Magic Quadrant for Security Information and Event Management, 2006 1H06' RA3 1192006.

[2] Sandeep K. B. The Operational Role of Security Information and Event Management Systems / K. B. Sandeep/ IEEE Security and Privacy Magazine, 2014. Vol. 12(5)— 35 pp.

[3] Effective Use Case Modeling for Security Information & Event Management [Електронний ресурс]. – URL: https://www.sans.org/reading - room/ whitepapers/ bestprac/paper/33319.

Main tasks of wireless network security supporting automated system

Joseph Kartvelishvili[1], Luka Shonia[2]

1. Faculty of Informatics and Management Systems, Georgian Technical University, Georgia Tbilisi, Kostava str. 77, e-mail: S.Kartvelishvili@gtu.ge

2. Faculty of Informatics and Management Systems, Georgian Technical University, Georgia Tbilisi, Kostava str. 77, e-mail: Lu.Shonia@gmail.com

Abstract - The paper describes the main tasks of an automated system supporting the wireless network security. Each task is designed to perform a certain function of the system and all of them are characterized by their functional purposes.

Keywords - Wireless Networks, Main Tasks, Information Packs, Automated System, Algorithms, Dialogue Procedures, Software Complex.

I. Introduction

Modern life has entered so particular stage of its development that any field requires for solution of practical tasks the development of effective methods and modern technical systems that provide easy, fast, and within a costly computer time to solve the set tasks.

It is associated with the development of computer and programming methods the security task of wireless networks, which requires the creation of an automated security system. This problem is a topical area of research and work is still being done to solve this problem.

II. Main material

The development of the main tasks of an automated system is initiated by examining the procedures for manual data processing in order to study and generalize the difficulties that a user may encounter. Such a study is of a general nature and is intended to identify difficulties rather than to determine their cause. Initially, a general overview of existing management procedures will be examined, followed by a separate study of each management task that is intended to be automated.

Security and quality of service in wireless networks have recently become a topic of great importance and a subject of active research, which is conditioned by a growing demand for data packet support. Without adequate security, organizations will avoid using wireless networks. Security issues in wireless networks are a major obstacle to widely adapting of such networks. Consequently, the security of such wireless networks is an important area that requires response if such networks are widely used. It is necessary that researchers of the field identify open problems and provide appropriate solutions to these problems. Each such attempt makes the wireless network slightly safer. The purpose of this paper is to develop a number of measures to enhance the security of wireless networks and to manage the remote workplaces.

According to the purpose above, the following issues are addressed in the paper:

- Enhance the security of routing information packets in wireless networks;
- Tightening of the mechanisms of encryption and authentication;
- Analysis of various communication channels in wireless networks and development of new methods to enhance security;
- Administration and management of remote workplaces (local or regional offices) using wireless networks;
- Remote control and administration of hardware settings used in the system;
- Processing and analysis of schedules for access to various local or remote facilities (at the level of the parties involved in a single system);
- Building a document turnover system (accounting for staff working hours, processing of access schedules, remuneration, etc.);
- Create and analyze histories at the level of each staff member;
- Creation of a unified information archive and statistical analysis based on it in the forms of tables and diagrams;
- Building and realization of automated system.

The main tasks structure of the security system is given in the Fig. 1.

Based on the main tasks of an automated security system, system algorithms need to be defined and developed, information processing and dialog procedures developed, and an automated system software complex created.

For each algorithm, input and output data should be identified, databases should be created and on its basis the security processes envisaged by the algorithms shall be carried out (Fig. 2).

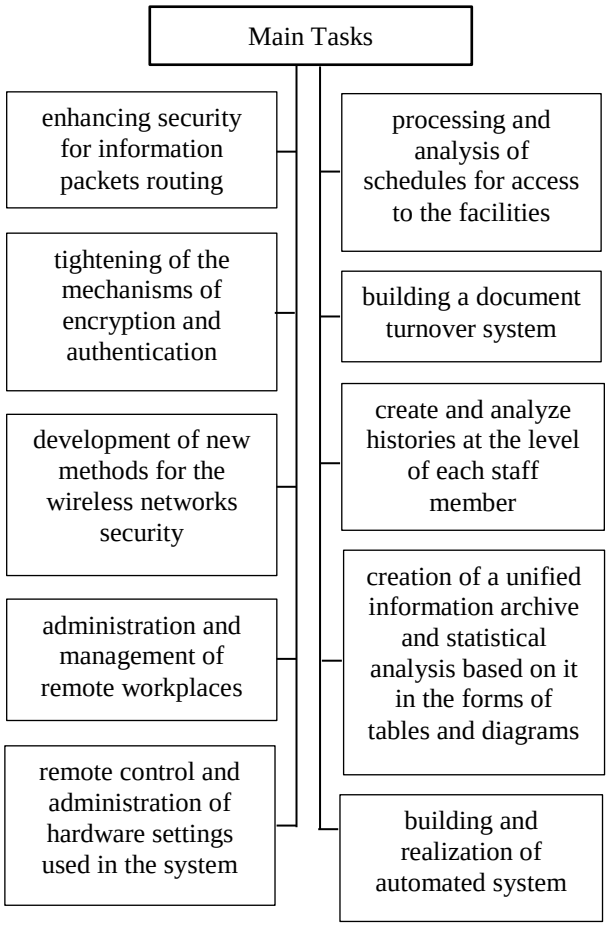


Fig. 1. Main tasks of the security system

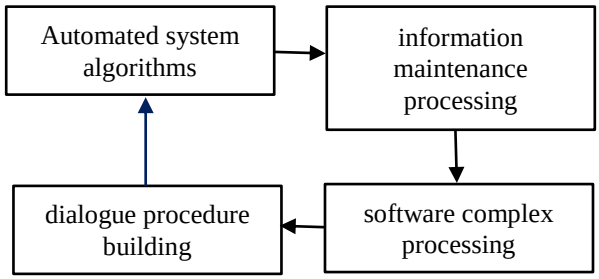


Fig. 2. Basic stages of designing an automated system

Dialogical procedures will be developed for the system, through which the organization of customer-system relations is carried out; these procedures are selected in such sequence that ensures that the certain functions are effectively implemented, in particular: creating of data set, updating, modifying, referencing, querying, processing, displaying, and more.

Conclusion

After performing these procedures, the automated system software complex and its structure will be processed. Software complex classes, constructors, procedures, functions and methods will be created.

Literature

[1]. O. Shonia, c. Nareshelashvili, I. Kartvelishvili. Wireless Network Security. Georgian Technical University, Tbilisi, 2009

[2]. Chogovadze G., Prangishvili A., Surguladze G. Book Hybrid Technologies and Data Management for Management Information Systems Programming. Georgian Technical University, Tbilisi, 2017.

Collect and Analyse Log Data with Cloud Services

Otar Shonia[1], Nino Topuria[2]

1. Faculty of Informatics and Control Systems, Georgian Technical University,
Kostava str.77,0160 Tbilisi, Georgia, Tbilisi,
E-mail:o.shonia@gtu.ge

2. Faculty of Informatics and Control Systems, Georgian Technical University,
Kostava str.77,0160 Tbilisi, Georgia, Tbilisi,
E-mail:nino.topuria@gtu.ge

Resume - The article proposes Microsoft cloud services, such as Azure Monitor and Power BI, for monitoring log data. Based on these services we can collect and aggregate azure monitor log data from a variety of sources into a common data platform, where it can be used for analysis, visualization, alerting. Power BI transforms log data into rich visuals, sharing on the web and mobile devices.

Keywords - azure, cloud, services, monitoring, analysis, visualization.

I. Introduction

Enabling observability across today's complex computing environments running distributed applications that rely on both cloud and on-premise services, requires collection of operational data from every layer and every component of the distributed system. You need to be able to perform deep insights on this data and consolidate it into a single pane of glass with different perspectives to support the multitude of stakeholders in your organization. [1],[2],[5]

II. Azure Monitor Views

Azure Monitor log contains different data types organized in records for different sets of properties for each type. Logs contain numeric values, such as Azure Monitor metrics, but usually contain textual data with detailed descriptions. They differ from these metrics in that they depend on their structure and are often not collected at regular intervals. Telemetry data, such as events and traces, is stored in Azure Monitor logs, additional performance data, so that it can be combined for analysis.

A common type of journal entry is an event that is regularly collected. Events are created through an app or service and usually contain enough information to provide context.

Since data formats may vary, applications create their own journal using the required structure. Metrics stored in tabs to combine with other monitoring data to analyze trends and other data.[3]

Log queries help us to fully leverage the value of the data collected in Azure Monitor Logs. Below is given log queris examples:

1. // Computers availability today

```
Heartbeat
| summarize dcount(ComputerIP) by bin(TimeGenerated, 1h)
| render timechart
```

Figure 1. Prezented Logs Chart example in Azure Monitor, where:
X axis - TimeGenerated [UTC]
Y axis - dcount_ComputerIP

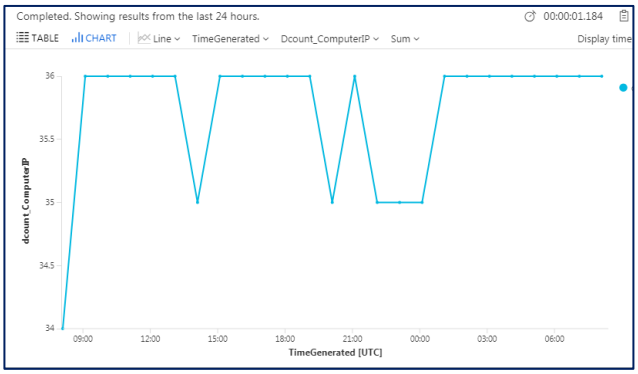


FIGURE 1. CHART THE NUMBER OF COMPUTERS SENDING LOGS, EACH HOUR

2. // Usage by data types

```
Usage
| summarize count_per_type=count() by DataType
| sort by count_per_type desc
| render piechart
```

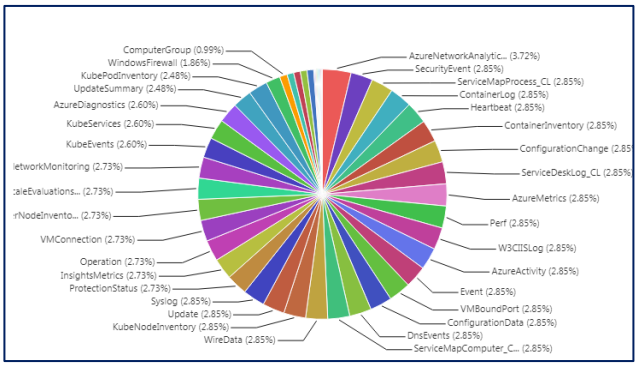


FIGURE 2. CHART THE AMOUNT OF LOGS REPORTED FOR EACH DATA TYPE,

We have:

- Rich visualizations for log data.
- Integrates into Azure Monitor management model with workspaces and monitoring solutions.
- Filters for custom parameters.
- Limitations:
- Supports logs but not metrics.
- No personal views. Available to all users with access to the workspace.
- No automatic refresh.
- Limited layout options.

– No support for querying across multiple workspaces or Application Insights applications.

III. Power BI dashbords

One of the features of Power BI Service is usage metrics report on a dashboard or report. The usage metrics report will give an analysis of how many times the content is viewed or share, through which platforms, and by which users [6].

So resourceful import the results of a log query into a Power BI dataset.

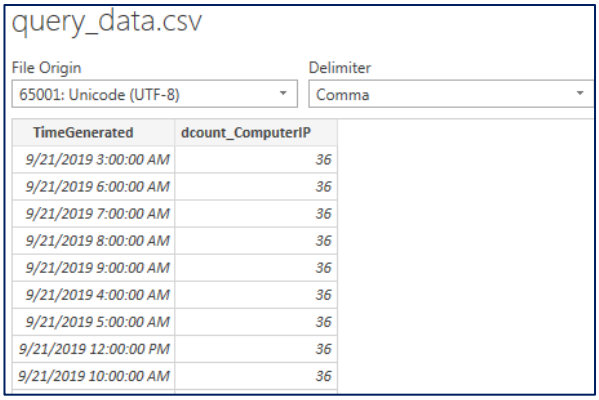


FIGURE 3. LOAD CVS-FILE IN POWER BI DESKTOP

Microsoft Power BI Desktop has various icons to represent different visualizations.[6] Depending on what we wish to display, we tick the checkbox for a particular data field, and then choose a chart type from the right pane.

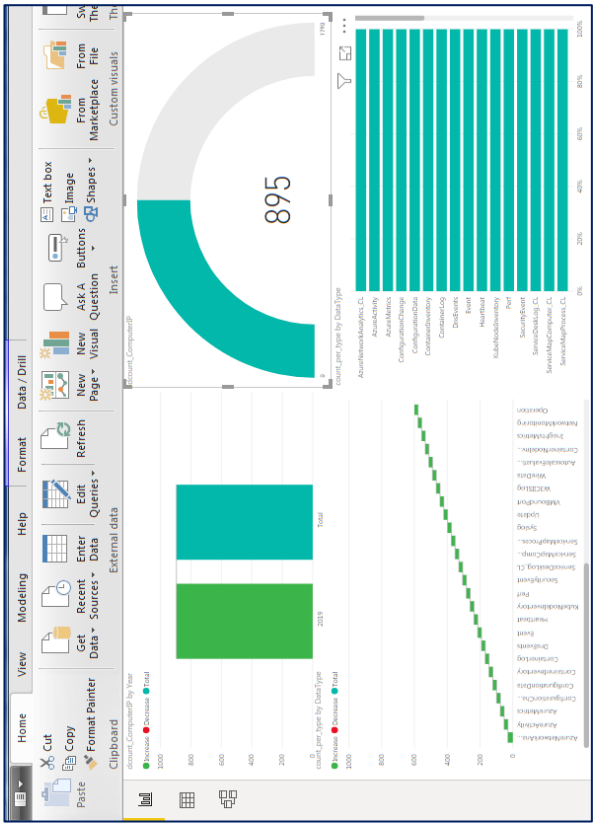


FIGURE 4. CHARTS IN POWER BI DESKTOP

In Power BI model we combining data from different sources.

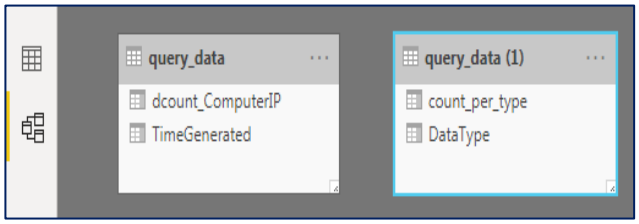


FIGURE 5. MAPPING TWO QUERY DATA IN A ONE MODEL

Now we use of Power BI Desktop to create reports, then publish those reports to the Power BI service, where we can view reports and dashboards.

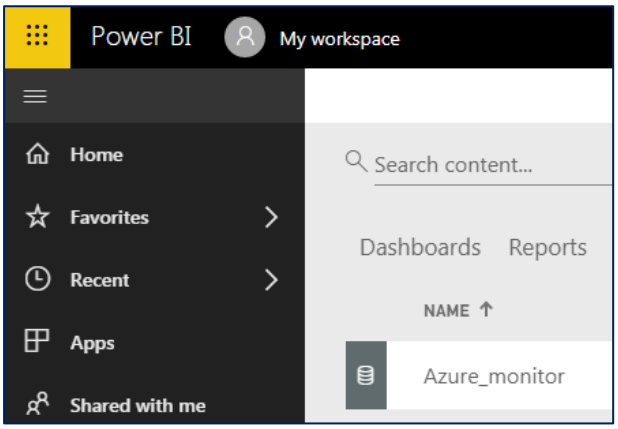


FIGURE 6 REPORT IN POWER BI SERVICE

IV. Conclusion

Azure Monitor log query and Power BI services take advantage of its features such as combining data from different sources and sharing reports on the web and mobile devices. Power BI give us rich visualizations and reports for analysis of different sets of log data.

[1] O.Shonia, N.Topuria / Local and cloud data protection tools.// Tbilisi, 2017, ISBN 978-9941-27-230-1.

[2] O.Shonia, N.Topuria /Realization of dynamic forms in the Internet portal of the corporation.// Modern information technologies in the context of economic globalization, International Conference N1, Tbilisi TSU, 2015

[3] S. Katamadze, N. Topuria/ Instant payment system business model and technical description // Computer Sciences and Telecommunications 2018, No.1(53), http://gesj.internet-academy.org.ge/en/list_aut_artic_en.php?b_sec=comp&list_aut=2788

[4] https://docs.microsoft.com/bs-latn-ba/azure/azure-monitor/platform/data-platform?view=vs-2017

[5] https://docs.microsoft.com/en-us/azure/azure-monitor/platform/data-platform-logs

[6] https://radacad.com/usage-metrics-or-do-it-yourself-power-bi-monitoring-report

Анализ проблем информационной безопасности в компьютерных сетях

Андрей Гавриленко, Владимир Караваев

Кафедра безопасности информационных технологий,
Харьковский национальный университет радиоэлектроники,
УКРАИНА, г. Харьков, пр. Науки, 14,
E-mail: andrii.havrylenko@nure.ua,
E-mail: volodymyr.karavaiev@nure.ua

Краткая аннотация – This article is devoted to information security in computer networks. The article discusses the security problems of computer networks and describes the recommendations for fixing vulnerabilities.

Ключевые слова – сеть, безопасность, кибербезопасность, политики безопасности, MikroTik, Ubiquiti, шифрование, VPN, IDS, Firewall.

I. Введение

Компьютеры и интернет стали неотъемлемой частью работы любой организации. Вся информация об организации, а так же результаты ее деятельности хранятся на компьютерах, что делает их интересными для злоумышленников, желающих каким-либо образом полчить выгоду из хранящейся на них информации.

Именно поэтому каждая организация должна контролировать свою информационную систему, чтобы обеспечить ее защиту и предотвратить возможные несанкционированные вторжения и атаки.

II. Актуальность проблемы

Когда компьютерная сеть не защищена должным образом, она становится уязвима для злонамеренного использования или случайного повреждения. Хакеры, недовольные сотрудники или плохая цифровая гигиена в организации может положительно повлиять на утечку персональных данных, включая коммерческие секреты и личные данные клиентов.

Например, потеря конфиденциальных исследований может потенциально обойтись организации в миллионы долларов, лишив ее конкурентных преимуществ. В то время как хакеры крадут данные клиентов и продают их для мошенничества, это создает негативную рекламу и общественное недоверие к организации.

Большинство распространенных атак на сети предназначены для получения доступа к информации, следя за коммуникациями и данными пользователей, а не для нанесения ущерба самой сети.

Но злоумышленники могут сделать больше, чем украсть данные. Они могут повредить устройства пользователей или манипулировать системами для получения физического доступа к объектам. Это оставляет имущество организации и членов группы под угрозой причинения вреда.

Грамотные процедуры обеспечения безопасности сети обеспечивают безопасность данных и блокируют уязвимые системы от внешних помех. Это позволяет пользователям сети оставаться в безопасности и сосредоточиться на достижении целей организации.

Более того, это означает, что клиенты и партнеры также могут уверенно взаимодействовать с организацией.

III. Решение проблемы

Ключевым этапом в обеспечении информационной безопасности сетей является грамотный подход к составлению политики безопасности сети.

Политика безопасности сети - это документ, котором прописаны рекомендации и методы обеспечения безопасности конкретной компании.

В конечном итоге для защиты сети необходимо реализовать различные уровни безопасности, чтобы злоумышленник должен был взломать две или более систем, другими словами, стоимость получения доступа к критически важным ресурсам, должна привышать стоимость этой информации.

Первым шагом в применении политик является определение политик, которые будут применяться. Сетевые политики определяют, как сеть должна быть реализована и настроена для оптимизации работы сотрудника, а также определяют, как реагировать при возникновении отклонений.

В контексте безопасности устройства, помимо разработки безопасности сети, разграничений доступа, компания так же должна подписать соглашение с каждым сотрудником о неразглашении сведений о развернутых устройствах внутри периметра, так же необходимо поддерживать ACL, чтобы разрешить или запретить трафик TCP и UDP, проводить регулярные исправления и обновления безопасности и т.д. [1].

Политики доступа в Интернет должны включать автоматическую блокировку всех веб-сайтов, которые были признаны неприемлемыми для пользователей компании. Кроме того, доступ в Интернет должен основываться на характере работы сотрудника.

VPN должен быть предназначен только для использования сотрудником принадлежащей организации компьютерной системы. Все виды удаленного доступа к корпоративной сети должны маршрутизироваться через VPN с действующей корпоративной лицензией, стандартной операционной системой и соответствующими исправлениями безопасности. [2]

Чтобы остановить возможное злоупотребление беспроводной сетью, должна быть обеспечена надлежащая аутентификация пользователя вместе с соответствующей заменой WEP и механизма отслеживания аномалий в беспроводной локальной сети. Кроме того, для шифрования должны использоваться меры безопасности 802.11i. [1]

IDS следует размещать для обнаружения аномалий и мониторинга несанкционированного доступа, так

как для экстремальной линии защиты брандмауэр или антивирус не достаточны. Администратор безопасности должен постоянно проверять файлы журналов системы и безопасности на предмет чего-либо подозрительного. Например, можно использовать Advance Antivirus, который имеет встроенную функцию IDS/IPS, для контроля несоответствующих прав аудита, повышенных привилегий, неправильных групп, измененных разрешений, изменений в реестре, неактивных пользователей и многого другого [3].

Данные, которые проходят по многим каналам, включая коммутатор, маршрутизаторы в сети в незашифрованном виде, уязвимы для многих атак, таких как спуфинг, переполнение SYN, сниффинг, изменение данных и перехват сеансов. Хотя вы не контролируете устройства, через которые могут передаваться ваши данные, но вы можете защитить конфиденциальные данные или защитить канал связи от доступа к данным в некоторой степени. Следовательно, использование многочисленных тактик шифрования, таких как SSL, TLS или IPSec, PGP, SSH, может зашифровать все виды связи, такие как POP, HTTP, POP3 или IMAP и FTP, поскольку пакеты SSL можно передавать через брандмауэры, серверы NAT и другие сетевые устройства без каких-либо особых соображений, кроме проверки наличия надлежащих портов на устройстве.

Определенная система или сервер, например, электронная почта, веб-сервер, база данных и т.д., которым требуется доступ к общедоступному Интернету, должны быть развернуты в выделенной подсети, которая отделена от внутренней системы извне, поскольку общедоступная система подвергается прямой атаке со стороны хакеров.

После составления политики безопасности сети, следует провести инструктаж всем пользователям сети. Так же, должен проводиться регулярный аудит политик безопасности.

Если коснуться технической реализации этого вопроса, то рынок предлагает массу вариантов, начиная от маршрутизаторов 3-4-го уровня, заканчивая системами предотвращения вторжений, сетевых фильтров от именитых производителей. Но чаще всего для руководства среднего и малого бизнеса остро стает вопрос цены обеспечения безопасности на предприятии, ведь, объективно, не все себе могут позволить покупать решения, например, тех же Cisco или Juniper, стоимость которых может достигать сотен тысяч долларов.

Но есть решения в более дешевом сегменте, с помощью которых можно реализовать практически все меры безопасности, описанные выше, это MikroTik и Ubiquiti - два известных конкурента в своем сегменте.

Среди специалистов нет однозначного ответа, что же лучше, MikroTik или Ubiquiti: все зависит от конкретного проекта, ведь если взять к примеру необходимость покрытия офиса точками доступа Wi-Fi и организацию бесшовного роуминга, то преимущество будет на стороне Ubiquiti, ведь точки доступа у них снабжены более мощными передатчиками и как показывает практика, даже при очень зашумленном эфире, они отлично справляются с передачей данных. С другой стороны, если для проекта важно не только покрытие беспроводной сетью, а еще и безопасная организация маршрутизации, настройка NAT, Firewall, создание безопасных VPN-тунелей между офисами организации, здесь, роутеры MikroTik имеют преимущество перед Ubiquiti.

Основные преимущества MikroTik: стоимость; функциональность - можно сравнивать с передовыми Cisco или Juniper; гибко конфигурируемая ОС; огромное количество документации и простое обновление; маштабируемость; единая операционная система RouterOS на всех устройствах.

К недостаткам можно отнести сложную настройку, ограничение пропускаемого трафика в 10 Гбит/с, так же отсутствует прямая гарантийная поддержка.

Выводы

В связи с ростом количества атак, осуществляемых на различные фирмы и корпорации по всему миру, сейчас, как никогда, возникает острая потребность в организации безопасного взаимодействия в сети компании. Для этого необходимо постоянно проводить аудит безопасности и вовремя устранять существующие потенциальные или реальные угрозы.

Но когда становится вопрос, с помощью чего же строить безопасность сети, зачастую ответ на этот вопрос лежит в платежеспособности фирмы и ценности охраняемой информации. Для больших корпораций, это однозначно Cisco, Juniper и другие известные бренды, что касается малого и среднего бизнеса, рекомендация остановиться на производителе MikroTik, как на самом функциональном и перспективном, в данной ценовой категории.

Литература

[1] Шаньгин В.Ф. Информационная безопасность компьютерных систем и сетей. / В.Ф. Шаньгин // ИД «ФОРУМ» - ИНФРА-М. – Москва, 2011.– С.383-384.

[2] Шаньгин В.Ф. Информационная безопасность компьютерных систем и сетей. / В.Ф. Шаньгин // ИД «ФОРУМ» - ИНФРА-М. – Москва, 2011.– С.300-306.

[3] Шаньгин В.Ф. Информационная безопасность компьютерных систем и сетей. / В.Ф. Шаньгин // ИД «ФОРУМ» - ИНФРА-М. – Москва, 2011.– С.258-263.

[4] Шаньгин В.Ф. Информационная безопасность компьютерных систем и сетей. / В.Ф. Шаньгин // ИД «ФОРУМ» - ИНФРА-М. – Москва, 2011.– С.343.

О некоторых подходах в DevSecOps к обеспечению безопасности ПО на основе процесса тестирования

Андрей Гапон[1], Владимир Федорченко[2]

1. Кафедра безопасности информационных технологий, Харьковский национальный университет радиоэлектроники, УКРАИНА, г. Харьков, пр. Науки, 14,
E-mail: gapon.andrei@gmail.com

2. Кафедра безопасности информационных технологий, Харьковский национальный университет радиоэлектроники, УКРАИНА, г. Харьков, пр. Науки, 14,
E-mail: fedorchenko.fedor@gmail.com

Annotation: The subject of research in the article is security approaches such as test driven development and penetration testing which are used in DevSecOps and provides proactive security for program applications.

Ключевые слова: DevSecOps, Test Driven Development, Penetration testing, программное обеспечение, реактивный подход, проактивный подход, белый ящик, черный ящик, серый ящик.

I. Введение

Уязвимости программного кода обычно появляются случайно на этапе разработки и внедрения программного обеспечения. Распространенные уязвимости включают ошибки проектирования, ошибки конфигурации, ошибки программного обеспечения и т. д.

DevSecOps меняет безопасность с реактивной на проактивную, при помощи различных подходов (методов), например, таких как Test Driven Development (разработка через тестирование) и Penetration testing (тестирование на проникновение) [1].

При реактивном подходе действие не начинается, пока не произойдет инцидент. После оповещения инициируется план реагирования компании на инцидент; вначале цель состоит в том, чтобы сдержать угрозу и восстановить службу.

При проактивном подходе организации пытаются обнаружить потенциальные угрозы до того, как произойдет инцидент. Данный подход также включает в себя регулярные запланированные учения для обнаружения угроз, которые могут скрываться в системе, но еще не обнаружены или, возможно, еще не активированы [2].

Контроль безопасности и тесты должны быть внедрены на ранних этапах и повсюду в жизненном цикле разработки, и они должны проводиться в автоматическом режиме.

Анализ проникновения зависит от двух методов, а именно от оценки уязвимости и тестирования на проникновение [3].

II. Penetration testing

Тестирование на проникновение – это санкционированная попытка отдельного лица или группы использовать существующие уязвимости в технической инфраструктуре организации и всех ее компонентах, чтобы определить, возможен ли несанкционированный доступ или злонамеренная деятельность [4].

Тестирование на проникновение классифицируются на основе уровня знаний и доступа. Спектр простирается от тестирования «черного ящика», когда тестер получает минимальные знания о целевой системе, до тестирования «белого ящика», когда тестеру предоставляется высокий уровень знаний и доступа. Этот спектр знаний делает различные методологии тестирования идеальными для различных ситуаций.

Тестирование «чёрного ящика» тестеру вообще не предоставляется никакой информации. Этот сценарий лучше всего подходит для определения того, как будет действовать злоумышленник, не обладающий знаниями о структуре приложения.

Тестирование «белого ящика» включает в себя предоставление полной информации о сети и системе тестеру, включая сетевые карты и учетные данные. Это помогает сэкономить время и снизить общую стоимость задания. Тест на проникновение в «белый ящик» полезен для моделирования целевой атаки на конкретную систему с использованием максимально возможного числа векторов атаки.

Тестирование «серого ящика» являет собой предоставление только ограниченной информации. Обычно это принимает форму учетных данных для входа. Тестирование «серого ящика» полезно и помогает понять уровень доступа, который может получить привилегированный пользователь, и потенциальный ущерб, который он может причинить. Тесты «серого ящика» устанавливают баланс между глубиной и эффективностью и могут использоваться для имитации внутренней угрозы или атаки, которая нарушила периметр сети [5].

На рис.1 приведены действия, которые необходимо выполнить для выполнения теста на проникновение

Рис.1 – Этапы выполнения теста на проникновение

По итогам тестирования на проникновение формируется отчет, который описывает обнаруженные уязвимости, а также степень их критичности и рекомендации по их ликвидации. На основании отчета принимаются меры по устранению выявленных уязвимостей.

III. Test Driven Development

Разработка через тестирование (TDD) – это подход, используемый для разработки кода на основе автоматизированных тест кейсов. Тестовая разработка включает в себя:

- Добавление теста, который фиксирует концепцию программиста о желаемом функционировании небольшого фрагмента кода.
- Запуск теста, который должен завершиться неудачей, так как код не существует.
- Написание кода и выполнение теста в тесном цикле, пока тест не пройдет.
- Рефакторинг кода после прохождения теста.
- Повторение этого процесса для следующего небольшого фрагмента кода, выполнение предыдущих тестов, а также добавленных тестов [6].

На рис.2 изображен цикл разработки программного обеспечения на основе подхода разработки через тестирование.

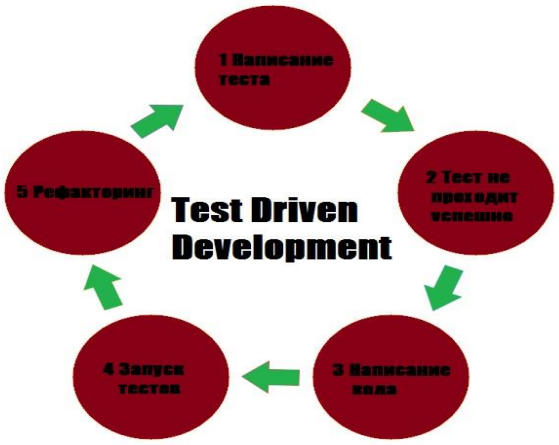


Рис.2 – Цикл разработки TDD

Разработка, основанная на тестировании, снижает вероятность появления дефектов, помогая разработчику сосредоточиться на четко определенных ожидаемых результатах. Испытания служат формой выполненной проектной спецификации для будущих работ по техническому обслуживанию. Тесты автоматизированы и используются при непрерывной интеграции.

Разработка через тестирование используется в гибких методологиях. Такой подход к разработке позволяет исправлять дефекты кодирования сразу после их появления.

Таким образом, при использовании подхода TDD, полностью предотвращается написание излишнего кода, поскольку, написание кода происходит исключительно для успешного прохождения тестов.

Выводы

Программное обеспечение и автоматизация продолжают изменять наш мир. Автоматизация в рамках жизненного цикла разработки программного обеспечения помогает нам быстрее и качественнее доставлять наш код. Добавление тестирования безопасности в эту автоматизацию также помогает создавать более безопасные приложения. DevSecOps – все еще новый подход и он развивается быстро.

В DevSecOps тестирование на проникновение должно выполняться на постоянной основе, чтобы идти в ногу с разработкой. Ручное выполнение тестов на проникновение может быть утомительной задачей, поскольку это может замедлить процесс разработки. И если это произойдет, следование принципам DevSecOps не даст никаких преимуществ.

Следовательно, существует острая необходимость в реализации автоматических тестов безопасности ПО для своевременного выявления недостатков, уязвимостей, утечки данных и лазеек.

Литература

[1] Guru99 [Електронний ресурс]. – Режим доступу : https://www.guru99.com/learn-penetration-testing.html

[2] DevSecOps Whitepaper [Електронний ресурс]. – Режим доступу : https://www.devseccon.com/wp-content/uploads/2017/07/DevSecOps-whitepaper.pdf

[3] DevOps [Електронний ресурс]. – Режим доступу : https://devops.com/shifting-data-protection-paradigm-proactive-vs-reactive/

[4] Breachlock [Електронний ресурс]. – Режим доступу : https://www.breachlock.com/penetration-testing-and-devops/

[5] Redscan [Електронний ресурс]. – Режим доступу : https://www.redscan.com/news/types-of-pen-testing-white-box-black-box-and-everything-in-between/

[6] *ISTQB* (*International Software Testing Qualifications Board*) [Електронний ресурс]. – Режим доступу : https://www.istqb.org/news/news/2014/189-in-chapter-3-1-1-in-agile-tester-extension-test-driven-development.html

Методика формального проектування КСЗІ в ІТС

Роман Гвоздьов[1], Володимир Заболотний[2], Артем Бойко [3]

1. Кафедра Безпеки інформаційних технологій Харківський національний університет радіоелектроніки, Україна, м. Харків, пр. Науки, 14,
E-mail: roman.hvozdov@nure.ua
2. Кафедра Безпеки інформаційних технологій Харківський національний університет радіоелектроніки, Україна, м. Харків, пр. Науки, 14,
E-mail: volodymyr.zabolotnyi@nure.ua

3. Приватне акціонерне товариство "Інститут інформаційних технологій", Україна м.Харків, вул. Бакуліна, 12,
E-mail: boyko@iit.kharkov.ua

Коротка анотація – Formal methods enable reasoning from logical or mathematical specifications of the behaviors of computing devices or processes; they offer rigorous proofs that all system behaviors meet some desirable property. They are crucial for security goals, because they can show that no attack strategy in a class of strategies will cause a system to misbehave. Without requiring piecemeal enumeration, they rule out a range of attacks. They offer other benefits too: Formal specifications tell an implementer unambiguously what to produce, and they tell the subsequent user or integrator of a component what to rely on it to do. Since many vulnerabilities arise from misunderstandings and mismatches as components are integrated, the payoff from rigorous interface specifications is large.

Formal methods differ from other design systems through the use of formal verification schemes, the basic principles of the system must be proven correct before they are accepted. Traditional system design has used extensive testing to verify behavior, but testing is capable of only finite conclusions.

Ключові слова – SDL, Z.100, LOTOS, E-LOTOS, UML, UMLsec, РІВЕНЬ ГАРАНТІЙ, ІТС, ФПБ

I. Вступ

Зі стрімким збільшенням кількості послуг, що надають інформаційно-телекомунікаційні системи (ІТС), зростає складність архітектури ІТС. Недоліки при проектуванні таких систем можуть критично вплинути на їх функціонування, зокрема на безпеку. Оскільки системи ускладнюються, безпека стає важливим питанням, формальний підхід до проектування системи пропонує ще один рівень страхування. Якщо в ІТС планується оброблення інформації, порядок захисту якої регламентується законами України або іншими нормативно-правовими актами (наприклад, інформації яка становить державну таємницю або вимоги до захисту якої встановлено законодавством), обов'язковим є незалежне підтвердження (оцінювання) відповідності реалізованих засобів та заходів захисту встановленим вимогам та нормам[1].

В Україні як критерії оцінки використовуються критерії, встановлені в [2]. Згідно з цими вимогами, оцінюються реалізовані функції захисту (функціональні послуги безпеки, ФПБ) та рівень гарантій коректності їх реалізації (рівень гарантій).

Рівень гарантій коректності реалізації ФПБ містять вимоги до архітектури комплексу засобів захисту (КЗЗ), середовища розробки, послідовності розробки, середовища функціонування, експлуатаційної документації та випробувань КЗЗ. Зокрема, вводиться сім рівнів гарантій (Г-1 ... Г-7), які є ієрархічними. Ієрархія рівнів гарантій відображає поступово зростаючу впевненість у тому, що реалізовані в об'єкті експертизи (ОЕ) ФПБ дозволяють протистояти певним загрозам, а також, що механізми, які їх реалізують, у свою чергу коректно реалізовані та можуть забезпечити очікуваний споживачем рівень захищеності інформації під час її оброблення в ОЕ [1].

Наприклад, для заявленого рівня гарантій Г-4 і більше реалізованої КЗЗ оцінюваного ОЕ необхідно викладати опис проекту архітектури у формалізованому вигляді, тобто використовуючи формальну нотацію. На даний момент часу не існує чітко визначеної методики для формального проектування КСЗІ в ІТС.

II. Вимоги до методики розробки

До методики розробки формального проектування КСЗІ в ІТС можна ввести наступні критерії:

1) орієнтованість на опис процесів обробки інформації;
2) орієнтованість на опис політики безпеки інформації;
3) однозначність та легкість сприйняття;
4) наявність готових блоків з безпеки.

III. Огляд методів формального проектування

SDL (Specification and Description Language) – мова специфікацій з формальною семантикою, призначена для опису телекомунікаційних систем. Стандарт мови визначений Міжнародним консультативним комітетом з телефонії і телеграфії (МККТТ) та включає рекомендації з Z.100 по Z.109. SDL має концепції поведінки, опису даних та структурування (особливо для великих систем) [2]. Основною опису поведінки є розширені скінчені автомати, що передають повідомлення. Опис даних ґрунтується на типах даних для значень та об'єктів. Основою структурування є ієрархічна декомпозиція та ієрархія типів [2]. Основна область застосування для SDL – конкретизація поведінки процесів в системах реального часу та розробка таких систем. До застосування у галузі телекомунікацій належать[2]:

1) обробка дзвінків та з'єднань (наприклад, обробка дзвінків, телефонна сигналізація, вимірювання) в комутаційних системах;
2) технічне обслуговування та усунення несправностей (наприклад, тривоги, автоматичне усунення несправностей, звичайні випробування) у загальних телекомунікаційних системах;

3) системний контроль (наприклад, контроль за перевантаженням, модифікацією та розширенням);

4) функції експлуатації та обслуговування, управління мережею;

5) протоколи передачі даних;

6) телекомунікаційні послуги.

Таким чином, SDL може використовуватися для опису:

а) вимог до об'єкта;

б) специфікацій системи;

в) технічних характеристик систем;

г) детальних технічних умов;

д) описи архітектури системи (як високого рівня, так і достатньо деталізованого для безпосереднього застосування реалізації);

е) описи системного тестування.

В сфері телекомунікацій SDL не має рішень з опису безпеки, а саме політик безпеки та готових блоків безпеки.

LOTOS (Language of Temporal Ordering Specification) – стандартизована мова специфікацій, призначена для опису комунікаційних та розподілених систем. LOTOS базується на CCS (Calculus of Communicating Systems, обчислення взаємодіючих систем) та CSP (Communicating Sequential Processes, комунікуючі послідовні процеси) для опису поведінки систем та на ACT-ONE для визначення абстрактних типів даних [3]. У LOTOS і E-LOTOS система, що підлягає специфікації, моделюється набором процесів, взаємодіючих між собою і їх оточенням. У LOTOS система описується сукупністю процесів. Процес може взаємодіяти з іншими процесами, що складають його середовище. Взаємодія між процесами базується на елементарних одиницях синхронізації, що називаються подіями або діями. Подія передбачає синхронізацію [4]: всі взаємодіючі процеси (два чи більше) беруть участь у події одночасно. Обмін даними може бути пов'язаний з цими синхронізаціями. Події є автономні, в тому сенсі, що їх виникнення відбувається миттєво, без тривалості. Вважається, що подія відбувається в точці взаємодії або в каналах подій (event gates). У разі синхронізації без обміну даними подія – це лише назва каналу подій[4].

UML (Unified Modelling Language) – мова графічного опису для об’єктного моделювання процесів. Модель UML складається з трьох основних категорій елементів моделі, кожна з яких може використовуватися для складання тверджень про елементи системи. Ці категорії [5]:

1) Класифікатор. Класифікатор описує набір об’єктів. Об'єкт - це елемент зі станом та стосунками до інших об’єктів. Стан об'єкта ідентифікує значення для цього об'єкта властивостей класифікатора об'єкта.

2) Подія. Подія описує набір можливих подій. Подія – це те, що відбувається, що має певний вплив на систему.

3) Поведінка. Поведінка описує набір можливих здійснень. Здійснення – це виконання набору дій (можливо протягом певного періоду часу), які можуть генерувати та реагувати на події, включаючи доступ та зміна стану об’єктів.

Також, існує UMLsec – це розширення до UML для інтеграції аспектів безпеки у специфікаціях UML. Опис аспектів безпеки реалізується за допомогою стереотипів (табл 3.1).

ТАБЛИЦЯ 3.1 – ПРИКЛАД UMLSEC СТЕРЕОТИПІВ (СКОРОЧЕНО)

Stereotype	Base Class	Description
Internet	link	Internet connection
encrypted	link	Encrypted connection
LAN	link, node	LAN connection
wire	link	wire
smart card	node	smart card device

Профіль безпеки UMLsec містить такі загальні ідеї[6]:

1) діаграма діяльності: безпечний потік управління, координація;

2) діаграма класу: обмін даними зберігає рівні безпеки;

3) діаграма послідовностей: взаємодія, яка критично важлива для безпеки;

4) схема діаграми стану: захищеність, що зберігається в об'єкті;

5) діаграма розгортання: вимоги фізичної безпеки;

6) пакет: цілісний погляд на безпеку.

Однак, UML з розширенням UMLsec не має в повному обсязі інструментів та визначень, які б дозволили в повному обсязі проектувати КСЗІ згідно з вимогами чинного законодавства.

Перераховані вище методи формального проектування, не задовільняють критеріям визначеними у пункті 2 цієї статті.

IV. Висновки

Актуальною задачею є розробка методики формального проектування КСЗІ в ІТС.

Література

[1] НД ТЗІ 2.7-010-09 «Методичні вказівки з оцінювання рівня гарантій коректності реалізації функціональних послуг безпеки в засобах захисту інформації від несанкціонованого доступу»

[2] ITU-T Recommendation Z.100 (11/99) Specification and description language (SDL)

[3] Lisandro Zambenedetti Granville and Maria Janilce Almeida. «Specification of E-LOTOS Systems in the E-DART Environment»

[4] Luc Leonard, Guy Leduc. «An introduction to ET-LOTOS for the description of time-sensitive systems»

[5] OMG Unified Modeling Language (OMG UML)

[6] Jan Jurjens, “TU Munich: UMLsec – Presenting the Profile 22 Requirements on UML extension for security II”

Селекція джерел радіовипромінювань декаметрового діапазону при використанні малогабаритних приймальних антен

Юрій Голобородько[1], Володимир Караваєв[2]

1. Державне підприємство “Центральне конструкторське бюро “ПРОТОН”, УКРАЇНА, м. Харків, Майдан Захисників України, буд.7/8, E -mail: proton@online.kharkov.ua; boroda_213@ukr.net

2. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, г. Харків, пр. Науки, 14, E-mail: volodymyr.karavaiev@nure.ua

The report discusses the selection of HF radio sources by type of radio waves. Aspects of source selection are considered when using small receiving antennas. The solution of the problem is based on sequential inspection of a given range with the thinning of the entire set of analyzed channels, taking into account the available a priori information about the specified radio emission.

Ключові слова – радіовипромінювання, частота, декаметровий короткохвильовий діапазон, оперативний радіоконтроль, частотний канал.

I. Вступ

Надійність та захищеність радіозв'язку в значній мірі залежить від завантаження використовуваного діапазону частот. Особливо це стосується діапазону декаметрових хвиль, що обумовлено особливостями їх розповсюдження [1]. В точку прийому надходять радіовипромінювання (РВ) як з ближньої, так і з дальньої зони. Під РВ ближньої зони розуміються РВ, що прийшли поверхневої електромагнітної хвилею, а під РВ дальньої зони розуміються РВ, що прийшли просторової хвилею (відбитої від іоносфери).

Для забезпечення надійності та захищеності радіозв'язку можливо використання оперативного радіоконтролю завантаження діапазону декаметрових хвиль [2].

II.Виявлення РВ

Для вирішення задачі оперативного радіоконтролю завантаження діапазону декаметрових хвиль необхідно здійснювати виявлення, визначення зони розміщення джерела РВ і розпізнавання заданих РВ, існуючих в заданому діапазон частот разом з безліччю невідомих РВ, що не представляють оперативного інтересу.

У декаметровому діапазоні хвиль існує безліч РВ, що розрізняються:

- частотою;
- розміщенням джерела в просторі (в ближній зоні - зоні поверхневого поширення електромагнітної хвилі і в далекій - зоні просторового поширення електромагнітної хвилі);
- видом радіовипромінювання (вид модуляції і маніпуляції, параметри сигналу, що модулює і т.д.);
- рівнем сигналу;
- часом появи;
- тривалістю.

У ряді випадків радіоконтролю діапазону представляють оперативний інтерес РВ, для яких апріорі відомі:

- вид радіовипромінювання;
- зона розміщення джерела радіовипромінювання (ближня);
- ділянки діапазону або номінали частот, на яких задані радіовипромінювання не можуть з'явитися.

Для оперативного вирішення поставленого завдання РТС повинна бути розміщена на рухомому носії, що накладає обмеження на обсяг апаратури і розміри АФС.

В даний час традиційно ця задача вирішується таким чином. За допомогою слухового або панорамного цифрового радіоприймального пристрою здійснюється пошук радіовипромінювань в заданому діапазоні частот. Потім всі виявлені випромінювання піддаються або слуховим, або візуально-апаратним, або автоматизованим способом аналізу для виявлення заданих радіовипромінювань.

Аналіз поставленого завдання показує, що компромісне рішення може бути отримано при послідовному огляді заданого діапазону з проріджуванням всієї сукупності аналізованих каналів з урахуванням наявної апріорної інформації про задані РВ. Зокрема, ряд частотних каналів немає необхідності аналізувати, оскільки в них задані РВ не можуть з'явитися. Крім того, можна виключити з аналізу частотні канали з радіовипромінювання далекої зони. Це дозволяє зменшити час послідовного огляду до величини, яка визначається заданими обмеженнями на час вирішення завдання (час огляду) виявлення і розпізнавання, які визначаються часом існування радіовипромінювання.

Потрібно при сформульованих умовах, введеному секторі показників якості та обмежених на показнику якості оптимізувати систему за сукупністю якості.

При цьому особливості розв'язуваної задачі наступні. Передбачається, що виробляється послідовний огляд декаметрового діапазону частот за допомогою автоматизованого панорамного комплексу РПК. При аналізі частотного каналу за сигналами з виходу панорамного приймача має прийматися рішення про наявність або відсутність одного із заданих РВ. Причому сигнали носять випадковий характер через випадкове в загальному випадку характеру переданих повідомлень, а також дію перешкод в частотному каналі. Тому завдання прийняття рішення в частотному каналі зводиться до

багатоальтернативного виявлення заданих випадкових сигналів при наявності невідомих сигналів. Для проріджування числа аналізованих каналів повинна бути використана апріорна інформація про характерні особливості заданих РВ.

За умовами завдання багатоальтернативність виявлення заданих РВ повинно проводитися в умовах підвищеної апріорної невизначеності. Зокрема, статистичні характеристики сигналів з виходу РПК апріорі не відомі. Однак можуть бути отримані навчальні вибірки сигналів для заданих РВ. Для всіх інших (невідомих, що заважають) РВ навчальні вибірки не можуть бути отримані, або вони є не представницькими, тобто недостатні для синтезу алгоритмів обробки.

При оптимізації структури алгоритмів виявлення повинна бути врахована сукупність показників якості виявлення і реалізованих витрат з урахуванням використання бортових засобів обчислювальної техніки. Реалізація засобами обчислювальної техніки забезпечить гнучкість структури отриманих обнаружителей.

Для скорочення (проріджування) числа каналів, в яких має здійснюватися багатоальтернативність виявлення заданих РВ, необхідно використовувати різні характерні ознаки РВ, засновані на обліку апріорних даних про задані РВ (ближня - дальня зона, номінал частоти і ін.) [6, 7]..

Для визначення зони розміщення джерела радіовипромінювання використовують або антенні системи з великою апертурою для вимірювання вертикальних кутів приходу електромагнітної хвилі (кутів місця), або непрямі методи, засновані на тривалому спостереженні за радіо випромінюванням (оцінка федингу, стійкості пеленга і т.п.) [3, 4, 5]. У зв'язку з тим, що при існуючих методах виявлення оператору пред'являються на аналіз всі виявлені РВ як задані, так і безліч невідомих (які заважають), швидкодія таких систем не задовольняє сучасним вимогам. Зокрема, час, що витрачається оператором на аналіз одного випромінювання, становить від одиниць секунд до одиниць хвилин. При такому способі виявлення і розпізнавання ймовірність виявлення короткочасних радіовипромінювань прагне до нуля, і для реалізації можливості виявляти короткочасні випромінювання доводиться різко знижувати ширину діапазону частот, що обслуговується одним оператором.

Для автоматизації процесу селекції РВ по зоні розміщення джерела пропонується використовувати відмінність в поляризаційних характеристиках поверхневої і просторової електромагнітних хвиль [8].

Висновок

Таким чином, запропонований метод селекції джерел радіовипромінювань декаметрового діапазону при використанні малогабаритних приймальних антен дозволяє в значній мірі підвищити надійність та захищеність радіозв'язку в даному діапазоні.

Література

[1] Комарович В. Ф., Сосунов В. М., Случайные радиопомехи и надежность КВ связи, М. «Связь», 1977, 135 с.

[2] К.т.н. Голобородько Ю. М., Кузниченко В. С. Перспективи розвитку засобів радіоконтролю. Збірник тез науково-практичної конференції „Проблеми забезпечення внутрішньої безпеки держави”, Харків, 2005.

[3] Обухов Н. П., Кикоть А. В., Голобородько Ю. Н., Горбачинский И. С., Авторское свидетельство № 177720 от 09.04.82 «Устройство автоматического панорамного обнаружения и пеленгования с дискриминацией радиоизлучений источников дальней или ближней зоны»

[4] Голобородько Ю. Н., Горбачинский И. С., Авторское свидетельство № 177874 от 01.09.82 «Панорамное приемное устройство с дискриминацией зоны размещения источника».

[5] Гурьев В. И., Горбачинский И. С., Голобородько Ю. Н., Авторское свидетельство № 210497 от 26.10.84 «Панорамное радиоприемное устройство для определения зоны нахождения источника радиоизлучения».

[6] Омельченко В. А., Голобородько Ю. Н. Многокритериальная задача многоальтернативного обнаружения сигналов в условиях повышенной априорной неопределенности, сообщение I. Радиотехника 1989, выпуск 90, Харьков, с. 28-35.

[7] К.т.н. В. В. Балабанов, к.т.н. Безрук В. М., к.т.н. Голобородько Ю. Н. Исследование алгоритмов обнаружения новых неизвестных радиоизлучений на фоне стационарной помехи. Збірник наукових праць ХВУ, випуск 5 (43), 2002

[8] Голобородько Ю. Н., Колесников Е. Н. Селекция источников радиоизлучений КВ диапазона по типу радиоволн при использовании малогабаритных приемных антенн. Сборник научных трудов 5-го Международного радиоэлектронного форума «Прикладная радиоэлектроника. Состояние и перспективы развития» (МРФ’2014), том 1 Конференция «Интегрированные информационные радиоэлектронные системы и технологии», Харьков, 2014, С. 1-294.

Аналіз ефективності протидії сучасних засобів захисту компаній HID-атакам

Ростислав Гриньов[1], Олександр Сєвєрінов[2]

1. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м Харків, пр. Науки, 14, E-mail: rost_grin@rambler.ru

2. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м Харків, пр. Науки, 14, E-mail: oleksandr.sievierinov@nure.ua

Коротка анотація – Interesting vector of attacks is use USB HID emulators of the keyboard (and mice) in the case of standard USB flash cards. And if autorun.inf on a flash card we already have learnt to search and destroy somehow with HID emulators all is meanwhile bad.

Комп'ютерні віруси, вразливість, HID-атака, антивірус, операційна система, утиліта.

I. Вступ

Питання безпеки в сучасних операційних системах не втрачає актуальності. Існує безліч різних векторів атак. Деякі з них вже давно відомі, деякі тільки з'явилися. Безмежна довіра операційних систем до таких пристроїв, як клавіатура або маніпулятор "миша" може нести загрозу безпеці. Якщо зібрати пристрій, який буде емулювати необхідне введення даних, і під'єднати його до комп'ютера, можна завдати серйозної шкоди системі.

Портативні носії даних дуже часто є джерелами поширення вірусного програмного забезпечення. Якщо раніше зловмисники використовували файл autorun.inf в корені флеш накопичувача, то останнім часом все частіше записують програму безпосередньо в мікроконтролер [1].

II. Можливі сфери застосування шкідливих hid-пристроїв

Сфери застосування такого запрограмованого мікроконтролера можуть бути різні, від застосування адміністраторами систем, фахівцями з безпеки під час проведення прихованого тесту на проникнення в компанії до використання такого пристрою зловмисниками. Якщо замаскувати подібний пристрій під виглядом маніпулятора "миша", клавіатури або флеш накопичувача, то можливо, що ним скористається хтось із співробітників. Відомі випадки, коли такі пристрої надсилались поштою в якості сувенірів або просто "губилися" поблизу організацій, наприклад, на парковці. Зазвичай, досить високий відсоток користувачів, які не замислюються про справжнє призначення пристрою і покладаються в даному питанні виключно на його зовнішній вигляд [2]. І тому дуже часто вони з упевненістю підключать подібне обладнання до комп'ютера. При цьому, ні система, ні, наприклад, антивірус не помічають вторгнення, оскільки визначають його як звичайну клавіатуру.

У подібних ситуаціях вектор атаки лежить на стику технології і соціальної інженерії. А саме, вимагає від потенційного зловмисника можливості фізично підключити пристрій, який визначиться як пристрій введення і самостійно виконає необхідні для нього дії.

HID, або Human Interface Device - тип комп'ютерного пристрою, який взаємодіє безпосередньо з людиною. Найбільш часто приймає від оператора вхідні дані і надає йому вихідні дані. Найпоширеніші типи HID-пристроїв - це клавіатура, маніпулятор "миша" і джойстик.

III. Засоби захисту сучасних компаній

Якщо розглянути комп'ютер користувача, то в якості суттєвих ознак, з точки зору забезпечення безпеки, необхідно враховувати: безпосередньо інформацію, що знаходиться в системі, антивірусне програмне забезпечення, брандмауер та засоби захисту операційної системи (захист компонентів ядра та системних файлів, розмежування доступу до файлів та ресурсів на підставі атрибутів, автентифікація та авторизація користувача). Однак ці засоби захисту не можуть виявити та протидіяти даній атаці. Якщо розглядати структуру організації, то до її суттєвих ознак можна віднести всі ознаки інформаційної моделі комп'ютера користувача а також: IDS, IPS, DMZ, сервер автентифікації та авторизації, пісочниці, honeypot, технологію розподілення та ізоляції мереж та інформаційних потоків на підставі типів та особливостей інформації, що циркулює. З точки зору засобів захисту та комп'ютерної системи HID-пристрої є повністю довіреними і в основному розглядаються як простий інтерфейс між користувачем і машиною, тому системи захисту організації не зможуть виявити таку атаку. Тому, коли до комп'ютера підключається нова клавіатура і маніпулятор "миша", система не запитує дозволу на їх установку, і драйвери найчастіше встановлюються автоматично. Така безмежна довіра може поставити під удар безпеку всієї системи.

Варто розуміти, що велика кількість витоків інформації з організації може відбуватись через неправильну утилізацію обладнання або під час ремонту. Наприклад, коли до ремонту потрапив комп'ютер, на жорсткому диску якого є фінансова звітність або розробки нового проекту. Проте правильна утилізація і виключення схожих ситуацій не гарантує безпеку. В будь-якій організації може виникнути ситуація, коли виходить з ладу устаткування. Це може бути мережевий пристрій, клавіатура. Після ремонту або заміни звичайного маніпулятора "миша" ніхто не помітить в ній наявності зайвого мікроконтролера, що може виконувати шкідливі дії. Такі атаки досить специфічні і мало розповсюджені, проте є найнебезпечнішими [3]. Оскільки в Україні мала кількість сертифікованих

сервісних центрів, то в найближчі декілька років ця проблема може стати досить поширеною. Послуги таких центрів можуть дорого коштувати, а з точки зору звичайної людини маніпулятор “миша” або клавіатура не можуть становити небезпеки для персонального комп’ютера або організації. Крім того, звичайний майстер, якого викликали виправити несправності, може встановити схожий пристрій.

IV. Типи та особливості шкідливих hid-пристроїв

Подібні апаратні закладки можуть бути досить різноманітними. Одні можуть мати бездротові інтерфейси, інші доступ через Інтернет, що дозволить зловмиснику під’єднуватися до них дистанційно [4]. Більш прості варіанти запрограмовані на виконання певних дій. Такі пристрої можуть бути приховані в системному блоці комп’ютера, маршрутизаторі, периферійному та іншому обладнанні. Небезпека атак, що використовують подібні пристрої полягає у важкості виявлення факту проникнення. Подібні пристрої можуть використовуватися зловмисниками для здійснення багатьох атак, бо можуть залишатися непоміченими роками.

За основу подібного шкідливого USB-пристрою ми візьмемо Arduino Leonardo Micro (рис. 1), так як дана плата має підтримку USB, достатню кількість пам’яті, невеликі габарити та низку вартість [5]. Прошивка може бути написана в середовищі розробки Arduino Development Environment. Саме з її допомогою можна редагувати і записувати програму в мікроконтролер. Розробка коду здійснюється за допомогою С-подібного синтаксису.

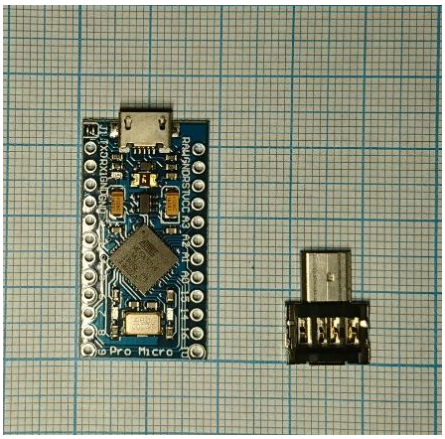


Рисунок 1 – мікроконтролер Arduino Leonardo Micro

Шкідливі пристрої подібного типу можуть виконувати різноманітні операції: від простого збору інформації про систему, викрадення паролів до повного контролю всієї системи і процесів з подальшим поширенням і зараженням нових пристроїв.

V. Захист від hid-атак

Існує кілька варіантів захисту від подібних атак:

1. Заборонити встановлення знімних пристроїв - це можна зробити за допомогою групової політики безпеки як для локальної машини, так і для робочих станцій в домені. Однак, при цьому буде не доступний Plug'n'Play.

2. Використовувати "білий список" - список довірених пристроїв. Слід врахувати, що пристрої ідентифікуються системою за рахунок ідентифікаторів Vendor ID і Product ID, які можуть бути запрограмовані зловмисником і повністю відповідати вже зареєстрованим в системі. Тому це не є абсолютним захистом.

3. Заборонити фізичний доступ до USB-портів. Проте таке рішення не є вдалим і може призвести до додаткових труднощів.

4. Використовувати для виявлення і блокування HID-емуляторів евристичні методи. Наприклад, ті, що грунтуються на аналізі зміни швидкості введення. Це найбільш раціональний і правильний підхід.

Висновки

У сучасному інформаційному світі питання захисту даних і безпеки стоїть дуже гостро, особливо при створенні захищених систем. Необхідно враховувати різні аспекти та вектори атак. Особливо ті, які базуються на соціальної інженерії і вразливості операційних систем. Важливо пам'ятати, що навіть такий пристрій, як звичайний флеш накопичувач, може нести серйозну загрозу не тільки для персональних комп'ютерів, але і для більш складних обчислювальних систем.

Література

[1] The fighting HID emulator - URL: http://developers-club.com/posts/141838/.

[2] Evil USB the HID-emulator or it is simple Peensy- URL: http://developers-club.com/posts/141838/

[3] Гриньов Р.С. Аналіз небезпеки впровадження вірусного програмного забезпечення в зображення / Р.С Гриньов, О.В. Севєрінов // Комп’ютерні та інформаційні системи і технології: між. науково-технічна конф. Харків, 2019. с. 75.

[4] Гриньов Р.С. Аналіз статистики та особливостей розповсюдження вірусів в Україні / Р.С Гриньов, О.В. Севєрінов // Сучасні напрямки розвитку інформаційно-комунікаційних технологій та засобів управління: між. конф. Харків, 2019. с. 100.

[5] Гриньов Р.С. Аналіз тенденцій вірусних загроз в Україні / Р.С Гриньов, О.В. Севєрінов // Сучасні напрямки розвитку інформаційно-комунікаційних технологій та засобів управління: між. конф. Харків, 2019. с. 120.

Организация оперативного управления кибербезопасностью на производстве

Алексей Демидов, Владимир Караваев

Кафедра безопасности информационных технологий,
Харьковский национальный университет радиоэлектроники,
УКРАИНА, г. Харьков, пр. Науки, 14,
E-mail: alexdemydov96@gmail.com,
E-mail: volodymyr.karavaiev@nure.ua

Abstract – Models of Security Operations Center are analyzed, their comparative characteristics are given. The basic components, tasks, processes and order of their deployment are considered. The practical value lies in developing recommendations for the enterprise to build Security Operations Center and to deploy them. The results of the research can be applied at Ukrainian enterprises. The scientific novelty lies in the factors that determine the model and architecture of Security Operations Center.

Keywords – SIEM; SOC; Security operations center.

I. Введение

В Украине за последние годы увеличилось количество кибер-атак в корпоративном секторе так и государственном, а сами хакеры уже не работают в одиночку, как 20 лет назад, а работают в крупных или не больших группах хорошо организованных как техническим оснащением так и огромными многомиллионными вложениями средств со стороны государств или средств которые были полученные незаконным путем.

Высокая интенсивность кибер-атак по всей Украине статистика приведена за 2019 год в Рис.1.

Рис. 1 - Статистика кибер-атак по Украине за 2019 год

При этом в корпоративном секторе работают специалисты по кибербезопасности, а в государственном секторе работает Киберполиция, СБУ ДКИБ, которые имеют в своем арсенале программно-технические комплексы в области защиты информации (DLP, SIEM, IDS/IPS, WAF/FW, EDR). При всей организации защиты информации в Украине, все равно происходят кибер-атаки с большим убытком государству.

Специалисты в сфере кибербезопасности давно осознали, то что необходимо создание единого централизированного комплексного решения в сфере реагирования на кибер-атак и других инцидентов связанных с информацией с возможностью расследования инцидентов. Основным решением является создание центра мониторинга и оперативного реагирования кибер-атак, который поможет избежать кибер-атак но и противостоять атакам в режиме реального времени а так же расследовать их после. Для обеспечения целостного и комплексного подхода мониторинга и реагирования на кибер-атаки, согласно всем стандартам регулирующим кибербезопасноть, например ISO IEC 27035, ISO IEC 27001. SOC объединяет все данные технологии а так же процессы и професиональные навыки сотрудников в сфере кибербезопасности, формируя комплексную систему защиты. Что позволяет получить высокую степень готовности и реагирования на инциденты в сфере кибербезопасности, что позволит избежать критических последствии от потенциальных кибер-атак, нацеленных на разные секторы в государстве.

II. Варианты SOC и их сравнение

Существуют три модели SOC:

- собственный SOC;
- SOC как сервис;
- гибридный SOC.

Их можно сравнить по следующим показателям:

- размещение технического оборудования;
- размещение персонала;
- уровень зрелости ИБ;
- структура расходов.

В Таблице 1. сравнивается место размещения технического оснащения.

Сравнивая эти типы можно выделить следующие преимущества и недостатки каждой из систем, которые приведены в сравнительной Таблице 2.

ТАБЛИЦА 1

Сравнение места размещения технического оснащения

	Собствен-ный SOC	SOC как сервис	Гибридный SOC
Оборудование для системы	*	+	*
Техническая поддержка	*	+	+
Серверы для сбора данных	*	*	*
Серверы для хранения данных	*	+	+
Серверы резервного копирования	*	+	+
Лицензирование (SIEM, Service Desk)	*	+	*

* - расположены в организации
\+ - расположены у интегратора

ТАБЛИЦА 2

Недостатки и преимущества различных типов SOC

	Собственный SOC	SOC как сервис	Гибридный SOC
Преиму-щества	Собственный процесс контролиру-емого самой организацией	Готовый процесс как услуга Сравнительно небольшая стоимость	Сравнительно небольшая стоимость готов процесс как услуга Возможность развернуть собственный SOC
Недос-татки	Высокая стоимость и сложность организации, построения и поддержки	Выход информации за рамки организации	Выход информации за рамки организации

III. Проблемы и их решение

Основные проблемы SOC:

- люди ;
- реагирование;
- обнаружение;
- расследование;
- поиск (Threat hunting);
- кадры.

Пути решения проблем SOC:

- использовать только лучшие решения SIEM для высокого уровня автоматизации, которые могут позволить создавать сложные правила корреляции и автоматизации с возможностью использования машинного обучения;
- обучение сотрудников, включая экспертов и специалистов в SOC;
- постоянный поведенческий анализ пользователей и систем таких как UBA и UEBA;
- использовать Threat Intelligence от вендоров и сообществ которые предаставляют Threat feeds и IOCs;
- постоянно искать новые API сервисы для интергации с SIEM;
- автоматизация повторяющихся инцидентов в SOC, для увелечения эффективности действий центра;
- постоянно искать лучшие варианты для улучшения политик и правил;
- чем выше осведомленность сотрудников, тем ниже загрузка аналитиков SOC.

Выводы

В статье рассмотрены варианты центров, их проблемы и решения. Были проанализированы их модели и предоставлена сравнительная характеристика. Центр реагирования на кибер-атаки может быть собственным или аутсорсинговым. В любом случае, внедряя SOC, организация одновременно реализует часть процессов системы управления информационной безопасности в соответствии со стандартом ISO 27001 процесс управления инцидентами ИБ, управления уязвимостями и изменениями, контроль соответствия законодательным и отраслевым требованиям, а также выполняет часть требований стандарта PCI DSS.

Литература

[1] S. David. A Practical Application of SIM/SEM/SIEM / Automating Threat Identification. SANS Institute, 2006. p.3. [Электронный ресурс]. - Режим доступа: https://www.sans.org/reading-room/whitepapers/logging/practical- application-sim-sem-siem-automating-threat-identification-1781.pdf

[2] Международный стандарт ISO / IEC 27001: 2013 «Система управления информационной безопасностью. Требования». [Электронный ресурс]. - Режим доступа: http://www.iso.org/iso/ru/catalogue_detail?csnumber=56742 .

[3] Интерактивная карта киберугроз. [Электронный ресурс]. - Режим доступа: https://cybermap.kaspersky.com/ru/

Особенности выбора базовых операций при построении симметричных блочных шифров

Роман Елисеев[1], Роман Олейников[2]

1. Кафедра безопасности информационных технологий,
Харьковский национальный университет радиоэлектроники,
УКРАИНА, г. Харьков, пр. Науки, 14,
E-mail: eliseev.roman371@gmail.com,

2. АО «ИИТ»,
УКРАИНА, г. Харьков, ул. Бакулина, 12,
E-mail: roliynykov@gmail.com

Block cipher algorithm can be created from wide range of operations. The main limitation – it must be present on all target platforms or implementer must be able to efficiently emulate it. Despite large amount of instructions in modern CPUs only few of them used in mainstream ciphers. Its addition, rotation, XOR and array (table) indexing mainly. But many of practical ciphers combines them with additional data manipulations to achieve better performance and/or security.

Ключевые слова – блочный шифр; операция; умножение; побитовая операция; возведение в степень; дискретный логарифм.

I. Введение

На сегодняшний день блочные шифры являются одним из основных средств защиты при передаче и хранении информации. Все современные стандарты проектируются с учетом возможности работы на широком спектре оборудования, однако существенные отличия между, например, смарт-картой, RFID меткой, смартфоном и сервером приводят к тому, что спроектировать одинаково подходящий всем им алгоритм сложно и приходится иметь дело с компромиссами.

Распространяются они как на объемы обрабатываемой информации, так и на доступность элементов, на основе которых может быть построен алгоритм.

Большинство современных государственных и международных стандартов базируется на использовании блоков подстановки в качестве основной нелинейной операции. Однако все большее распространение получают алгоритмы на основе простых арифметических и логических операций процессоров.

Связано это в первую очередь с тем, что такие шифры обычно производительнее и компактнее в программной реализации, но при этом могут быть достаточно просто реализованы в аппаратном исполнении.

При этом разные платформы могут иметь различный уровень поддержки математических операций. Так, например, современные процессоры общего назначения имеют наборы векторных инструкций, позволяющих за один раз производить операции над несколькими наборами операндов, в то время как некоторые микроконтроллеры не имеют даже аппаратной поддержки умножения.

II. Обзор операций

Среди операций, не нашедших широкого применения в симметричных блочных шифрах, можно выделить умножения по модулю и побитовые вращения на переменное количество бит. Хорошим примером шифра, который использует обе операции является RC6[1], признанный одним из самых простых и красивых с математической точки зрения алгоритмов. Его предшественник, RC5[2] (Рис. 1), использует побитовые вращения на значения, зависящие от данных. К известным «пользователям» модульного умножения относится и IDEA (Рис. 2) [3].

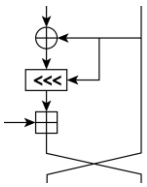


Рисунок 1. Раундовая функция RC5

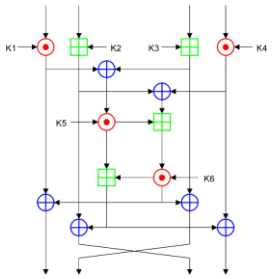


Рисунок 2. Раундовая функция IDEA

Обе эти операции сложны для классического криптоанализа и поэтому могут рассматриваться как перспективные при комбинировании с «основными», о чем говорит история IDEA [4], но в то же время они сложны и для реализации на некоторых типах устройств (некоторые виды маломощных микроконтроллеров, смарт-карты, RFID метки).

Связано это в первую очередь с тем, что некоторые узкоспециализированные архитектуры могут не включать операции умножения. Так же, в процессорах общего назначения операция умножения несколько медленнее простых арифметических операций [5].

На ряду с ней возникает проблема с атаками по времени выполнения [6], что связано в первую очередь с попытками процессора оптимизировать тяжелую операцию в среднем случае. Все эти проблемы касаются и побитовых сдвигов [7].

Другим интересным примером является SAFER [8] (Рис. 3), блоки подстановки которого основаны на возведении в степень и дискретном логарифме по модулю 257.

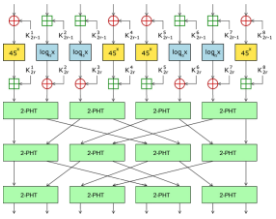


Рисунок 3. Раундовая функция SAFER

В случае же с этим шифром, проблемы возникают не на 8-битных микроконтроллерах (при условии предварительно рассчитанных таблиц), а на всех остальных платформах, так как все операции в шифре оперируют байтами и использование более широких слов не дает преимуществ, а часто еще и замедляет алгоритм в силу ручных приведений по модулю (при отсутствии 8-битных регистров в архитектуре). Кроме того, полученные таким методом блоки подстановки уступают псевдослучайным с точки зрения стойкости к ряду атак.

К малораспространенным на данный момент операциям в шифрах можно отнести и побитовые операции И и ИЛИ, а также нециклические сдвиги. Примерами шифров, использующих операцию И, являются RC2[9], SIMON[10] (Рис. 5), MISTY1[11]. Последний алгоритм использует так же и ИЛИ.

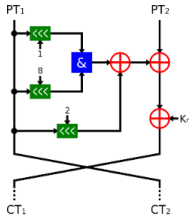


Рисунок 5. Раундовая функция SIMON

Одними из алгоритмов, которые используют побитовые нециклические сдвиги, являются SERPENT [12] и XTEA [13].

Преимуществами побитовых операций является их простота и доступность, а к недостаткам можно отнести низкий лавинный эффект [14], что приводит к необходимости использования большего количества операций по сравнению с, например, более сложным, но в то же время примерно таким же быстрым (на программных платформах), сложением по модулю 2^n.

Выводы

Использование сложных операций в блочных шифрах имеет как преимущества в виде повышения производительности на некоторых платформах или повышения стойкости к отдельным типам атак, но также и недостатки вроде появления новых векторов атак (например, атак по времени выполнения) и усложнения реализации на некоторых платформах.

Так же это усложняет криптоанализ алгоритма в силу меньшего количества уже известной информации об операциях, их взаимодействии с другими операциями.

Литература

[1] R.L. Rivest, M.J.B. Robshaw, R.Sidney, and Y.L. Yin. The RC6 Block Cipher. v1.1, August 1998.

[2] Rivest, R. L. (1994). "The RC5 Encryption Algorithm" (pdf). Proceedings of the Second International Workshop on Fast Software Encryption (FSE) 1994e: 86–96.

[3] Xuejia Lai and James L. Massey, A Proposal for a New Block Encryption Standard, EUROCRYPT 1990, pp. 389–404

[4] Khovratovich D., Leurent G., Rechberger C. (2012) Narrow-Bicliques: Cryptanalysis of Full IDEA. In: Pointcheval D., Johansson T. (eds) Advances in Cryptology – EUROCRYPT 2012. EUROCRYPT 2012. Lecture Notes in Computer Science, vol 7237. Springer, Berlin, Heidelberg

[5] 2001. Performance Optimization of Numerically Intensive Codes. Soc. for Industrial and Applied Math., Philadelphia, PA, USA.

[6] Kelsey J., Schneier B., Wagner D. (1996) Key-Schedule Cryptanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES. In: Koblitz N. (eds) Advances in Cryptology — CRYPTO '96. CRYPTO 1996. Lecture Notes in Computer Science, vol 1109. Springer, Berlin, Heidelberg

[7] Handschuh H., Heys H.M. (1999) A Timing Attack on RC5. In: Tavares S., Meijer H. (eds) Selected Areas in Cryptography. SAC 1998. Lecture Notes in Computer Science, vol 1556. Springer, Berlin, Heidelberg

[8] Massey J. SAFER K-64: A byte-oriented block-ciphering algorithm // Fast Software Encryption: Cambridge Security Workshop Cambridge, U. K., December 9–11,1993 Proceedings / R. J. Anderson — Berlin: Springer Berlin Heidelberg, 1994. — P. 1–17. — (Lecture Notes in Computer Science; Vol. 809) — ISBN 978-3-540-58108-6 — ISSN 0302-9743 — doi:10.1007/3-540-58108-1_1

[9] R. Rivest, A Description of the RC2(r) Encryption Algorithm (RFC2268, Informational), March 1998

[10] Simon and Speck: Block Ciphers for the Internet of Things (англ.). NIST Lightweight Cryptography Workshop (9 July 2015).

[11] RFC2994 A Description of the MISTY1 Encryption Algorithm. H. Ohta, M. Matsui. November 2000. (Format: TXT, HTML) (Status: INFORMATIONAL) (DOI: 10.17487/RFC2994)

[12] Ross Anderson, Eli Biham, Lars Knudsen. Serpent: A Proposal for the Advanced Encryption Standard

[13] Tom St Denis. Extended TEA Algorithms

[14] Ramanujam S., Karuppiah M. Designing an algorithm with high Avalanche Effect //IJCSNS International Journal of Computer Science and Network Security. – 2011. – Т. 11. – №. 1. – С. 106-111.

Дослідження криптосистем на основі кодів хеш функцій

Ілля Жеков, Володимир Караваєв

Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА,
м.Харків, пр. Науки, 14,
E-mail: illia.zhekov@nure.ua,
E-mail: volodymyr.karavaiev@nure.ua

Коротка анотація – This article is devoted to information security in post quantum epoch. The article discusses the security problems which could appear with quantum computer appearing and describes what have world's best minds do to prevent information's collapse.

Ключові слова – квантовий комп'ютер, хеш-функція, алгоритм Шора, дискретний логарифм, NIST, криптоаналіз, криптосистема, стандартизація.

I. Вступ

В наш час багато провідних учених і експертів активно працюють над створенням квантових комп'ютерів. Кожний додатковий кубіт у два рази збільшує площу пошуку даних, отже, значно підвищується й швидкість їх обчислення.

Квантові комп'ютери ймовірно зможуть зруйнувати більшу частину, якщо не абсолютно всі традиційні криптосистеми, які широко використовуються в практиці. Конкретно, системи, засновані на завданні факторизації цілих чисел (наприклад, RSA).

Саме тому вже зараз ведеться робота над створенням криптосистем стійких до квантового криптоаналізу.

II. Актуальність проблеми

На сьогоднішній день майже всі технології в мережі інтернет містять у собі асиметричне шифрування. Це передбачене необхідністю мати захищений канал передачі між вузлами[1].

Мільярди людей щодня використовують протокол TLS коли вони входять в акаунти соціальних мереж, форумів, електронної пошти або будь-якого іншого ресурсу де має місце структура логін-пароль. Це вже не говорячи про технологію електронного цифрового підпису, який широко використовується у фінансовій сфері й на рівнях державних таємниць.

Також користувачі сучасних комп'ютерів, смартфонів досить часто завантажують нові програми для своїх гаджетів. Про це може свідчити величезний ринок цифрових продуктів, який тільки продовжує рости. Але для завантаження будь-якого додатка або програми, що використовують інтернет при роботі, потрібне залучення технологій шифрування.

Тому криптографія з відкритим ключем і тут є невід'ємною частиною. У зв'язку із цим має місце постійна необхідність підтвердження надійності й відмовостійкості сучасних криптографічних систем.

Достовірність будь-якого файлу, що завантажується з інтернету повинна бути підтверджена за допомогою цифрового підпису, щоб ви могли бути впевнені в тому, що ніде не була порушена цілісність. Така структура впроваджена у всіх ведучих постачальників електронних продуктів таких як: Appstore, Googleplay, Amazon, Tesla, Microsoft, Ubuntu і т.д.

Втративши свій рівень безпеки, будь-який файл можна вважати скомпрометованим і неприпустимим для використання, тому провідні компанії постійно поліпшують свої продукти, надаючи людям упевненість у безпеці. Одним з варіантів підвищення якості послуг безпеки є використання сучасних криптосистем, і зокрема на основі кодів хеш функцій.

III. Аналіз проблеми

В 1994 році Шор [3] запропонував квантові алгоритми дискретного логарифмування в групі точок еліптичної кривої і факторизації чисел. В 2001 році в IBM продемонстрували працездатність алгоритму Шора, розклавши число 15 на множники 3 і 5 на 7-кубітному квантовому комп'ютері. По оцінці Proos і Zalka відновлення секретного ключа ECDSA довжиною 256 біт зажадає близько 1500 кубіт і 6*109 (~232) операцій. По оцінці фахівців Microsoft для цього буде потрібно 2330 кубіт і $1.26*10^{11}$ ($\sim 2^{36}$) операцій.

У звіті Національного Інституту Стандартів і Технологій США (The National Institute of Standards andTechnology, NIST) за квітень 2016 року відзначається, що більшість асиметричних криптографічних примітивів, широко використовуваних сьогодні в різних сферах суспільного життя, і які базуються на завданнях факторизації та дискретного логарифмування в різних групах, будуть скомпрометовані.

У силу всього вище сказаного, очевидною стає необхідність подальшого розвитку пост-квантової криптографії. Тому що схеми електронного підпису повністю втратять свою криптостійкість у випадку появи квантового комп'ютера, на відміну від шифрування й обміну ключами, саме для цих криптографічних функцій першорядною необхідністю є пошук нових пост-квантових аналогів.

Пост-квантова криптографія на даний момент містить у собі наступні основні підходи:теорія ґрат; багатомірні квадратичні системи; електронні підписи на хеш-функціях; теорія алгебраїчного кодування; ізогенії еліптичних кривих.

Розглянемо коротко переваги й недоліки кожного підходу, приведемо приклади конкретних реалізацій.

Криптографія на ґратах. Даний розділ криптографії почав активно розвиватися з 1990-х років і містить у собі велику кількість важко обчислювальних завдань, деякі з яких уважаються Np-повними . Більшість схем прості в розумінні, забезпечують гарну швидкодію й мають властивість розпаралелювання обчислень. Крім шифрування й підпису, на ґратах можуть бути побудовані інші цікаві додатки (повністю гомоморфне шифрування, шифрування й підпис із використанням атрибута, обфускація кодів і інші). Деякі системи із цього розділу мають складність у

найгіршому випадку, а не в середньому, як більшість криптосистем. До мінусів можна віднести відсутність точного методу оцінки складності алгоритмів на ґратах до існуючих видів атак. Найбільш відомою схемою є криптосистема NTRU (Nth-degree Truncated polynomial ring), запропонована в 1998 році. На базі криптосистеми NTRU можна реалізувати алгоритми шифрування й електронному підпису.

Модифікована версія даного алгоритму була взята за основу стандарту для фінансових організацій ANSI X9.98-2010 «Lattice-Based Polynomial Public Key Establishment Algorithm for the Financial Services Industry». В 2008 році криптосистема NTRU була включена в стандарт IEEE 1363.1 «Lattice-based public-key cryptography».

Криптографія, заснована на багатомірних квадратичних системах. Стійкість цього розділу криптографії ґрунтується на складності розв'язку системи багатомірних квадратичних багаточленів над кінцевим полем. Дане завдання вважається Np-повним. Системи із цього розділу мають гарну швидкість і невеликі вимоги до обчислювальних ресурсів, однак, довжини відкритих ключів досить великі.

Криптографія на кодах, що виправляють помилки (теорія алгебраїчного кодування). До плюсів такого роду систем можна віднести швидкість обчислень. До мінусів - занадто велику довжину ключів. На теорії алгебраїчного кодування базуються класичні криптосистеми McEliece і Niederreiter.

Ізогенії суперсингулярної еліптичної кривої. Найбільш популярний протокол SIDH (Supersingular isogeny Diffie-Hellman, SIDH) дозволяє зробити обмін ключами по незахищеному каналу зв'язку. Цей факт і є його відмінною рисою, що гарантує досконалу таємність. З урахуванням стиснення SIDH має найменшу довжину ключа із усіх постквантових протоколів обміну ключами. Однак повноцінної криптосистеми на ізогеніях поки реалізовано не було.

Криптографія, заснована на хеш-функціях з нашої точки зору є досить перспективним напрямом. У даний розділ входять електронні підписи, побудовані за допомогою хеш-функцій, у силу чого забезпечується їхня стійкість до квантових обчислень.

IV. Розв’язання проблеми

Криптостійкість цифрового підпису, заснованого на гешуванні, зводиться до стійкості хеш-функції до відновлення першого й другого прообразу. Відновлення прообразу хеша довжиною n біт на класичному комп'ютері зводиться до повного перебору (складність $O(2^n)$), на квантовому — до перебору алгоритмом Гровера (складність $O(2^{n/2})$). Таким чином, можна вибрати стійку хеш-функцію, що забезпечує необхідну класичну й пост-квантову криптостійкість. Інші підходи до постквантової криптографії ґрунтуються на математичних проблемах, які вважаються квантово-стійкими, але можливо можуть бути вирішені класичним комп'ютером у випадку прориву в розвитку математики.

Згідно з рекомендаціями PQCRYPTO для досягнення 128-бітної постквантової криптостійкості на практиці може використовуватись схема цифрового підпису XMSS [4,5] з параметрами з RFC 8391. Недолік схеми XMSS полягає в тому, що вона може генерувати обмежене число підписів, яке залежить від висоти використовуваного дерева Меркла.

Поряд з XMSS рекомендації PQCRYPTO містять у собі схему SPHINCS [6], яка позбавлена обмеження на кількість підписів, але для 128-бітної криптостійкості розмір такого підпису становить близько 41 кілобайт, тому більшою мірою цей алгоритм придатний для використання в системах автентифікації.

В цій схемі замість OTS (One Time Signature) використовується FTS (Few Time Signature), що дозволяє зменшити ймовірність виникнення колізії шляхів та зменшити висоту дерева. По-друге, внутрішні вузли дерева замінюються деревами Мерклє. За допомогою використання такої конструкції зменшується необхідний на генерації підпису час та розмір підпису, адже до самого підпису входить менша кількість реалізацій OTS.

Висновки

У даній роботі був проведений аналіз існуючих пост-квантових підходів, відзначені переваги й недоліки даних підходів. Також були розглянуті пост-квантові схеми криптосистем на основі геш-функцій, одна з них SPHINCS пройшла в другий тур конкурсу NIST на створення нових пост-квантових стандартів. Проведений порівняльний аналіз даних кандидатів. Нові стандарти підсилять FIPS 186-4, стандарт цифрового пыдпису (DSS), а також 800-56A.

Література

[1] PQCrypto 2017. Netherlands, 26-28 June 2017. [Електронний ресурс] – Режим доступа: URL: https://2017.pqcrypto.org/conference/ - 02.06.2018.

[2] Bernstein D. J., Buchmann J., Dahmen E.: Post-Quantum Cryptography. — Springer. — 2009.

[3] Shor P. W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer//Foundations of Computer Science: Conference Publications. — 1997. — P. 1484–1509.

[4] PQCRYPTO. Initial recommendations of long-term secure post-quantum systems. http://pqcrypto.eu.org /docs/initial-recommendations.pdf.

[4] J. Buchmann, E. Dahmen, A. Hülsing. XMSS – A Practical Forward Secure Signature Scheme based on Minimal Security Assumptions. https://eprint.iacr.org /2011/484.pdf

[5] SPHINCS: practical stateless hash-based signatures. http://sphincs.cr.yp.to/

Порівняння перспективних постквантових алгоритмів ЕП на базі MQ-перетворень

В.С. Засипко[1]

1. Кафедра безпеки інформаційних систем і технологій, Харківський національний університет імені. В. Н. Каразіна, Україна, м. Харків, площа Свободи 6
vladislavzasypko@gmail.com

Коротка анотація - Стаття має за мету базово розглянути кандидатів конкурсу постквантових алгоритмів NIST, що базуються на MQ-перетвореннях. У статті наведені основні та найбільш значущі дані про кожну зі схем, їх переваги і недоліки. Також виділені підсумки та загальне порівняння цих алгоритмів.

Ключові слова - NIST PQC, електронний підпис, LUOV, MQDSS, GeMSS, Rainbow.

I. Вступ

Національний Інститут Стандартизації та Технологій (NIST) наразі стоїть перед вибором найкращих криптографічних алгоритмів для постквантового періоду. Відбір проходить у вигляді конкурсу відкритого типу.

Серед алгоритмів ЕП на базі MQ-перетворень у другий етап вийшли 4 алгоритми: LUOV, MQDSS, Rainbow, GeMSS. З особливостей такого виду методів ЕП є невеликі складність асиметричних перетворень та ресурси, що потрібні для здійснення перетворень.

II. LUOV

Схема LUOV є адаптацією схеми підпису UOV. Вона відрізняється від прототипу в декількох напрямках:

- новий алгоритм дозволяє обрати більшу частину відкритого ключа;
- відкритий ключ є «піднесеним» до роз-шireного поля F_2 шляхом розширення поліноміального відображення від F_2^n до F_2^m;
- в алгоритмі підпису змінні визначено генеруються на основі повідомлення та закритого ключа [1].

Матричне представлення лінійного відображення T робить алгоритми генерації ключа та підпису значно швидшими:

$$T = \begin{pmatrix} 1_v & T \\ 0 & 1_m \end{pmatrix} \quad (1),$$

де T – це (v*m) матриця.

Алгоритм є консервативним в плані безпеки і припускає, що квантові атаки проти багатомірних підписів можуть поліпшитися, цим самим гарантуючи, що алгоритм зможе пережити незначні удосконалення криптоаналізу [2].

Схема використовує псевдовипадковий генератор для побудови частини відкритого ключа, для якого може бути вирішена відповідна частина секретного ключа, подібно до схеми циклічного UOV, циклічного Rainbow та їх псевдовипадкових аналогів. LUOV пропонує явний компроміс між розміром ключа та довжиною підпису, що робить схему гнучкою для різнорідних випадків використання. Сума розміру відкритого ключа і довжини підпису є найменшою для схем багатоваріантного підпису кандидата.

Перевагами алгоритму є високий рівень криптографічної стійкості, мала довжина відкритого та секретного ключів, а також відносно мала довжина підпису.

III. MQDSS

Це перша серед багатовимірних схем підпису, чиї показники захищеності були доведені у моделі випадкового оракула. Механізм розроблений шляхом застосування до 5-крокової схеми ідентифікації перетворення Фіата-Шаміра (Fiat-Shamir transformation, FST). На конкурс були запропоновані два набори параметрів на 1-2 та 3-4 рівні стійкості.

Схема довела EU-CMA захищеність, тому «кращими» атаками на криптосистему є атаки на псевдовипадковий генератор та на геш-функцію, а оскільки вони базуються на SHAKE256, то зрештою це будуть атаки на алгоритми гешування. Також проблемою виступає атака на задачу MQ, а загальні алгебраїчні методи мають найвищу загрозу для MQ в схемі MQDSS, а отже і для усієї системи [3].

Перевагами алгоритму є:

- Малий розмір ключів;
- «Гнучкі» параметри, що можна підлаштовувати до різних платформ та рівнів стійкості.

Недоліки:

- Великий розмір підпису;
- Підтверджена захищеність у ROM, але не у QROM. До того ж від захищеності у ROM страждає розмір підпису (приблизно 120 KB).

IV. Rainbow

Схема була запронована в 2000-х роках та базується на алгоритмі UOV. Є одним з найстарших кандидатів, а отже і найбільш дослідженим. На конкурс була запропонована вдосконалена схема, що гарантує модель безпеки EUF-CMA до 2^{64} обраних повідомлень. Сутність змін полягає у додаванні «солі» до повідомлення, яке гешується. Для того, щоб отримати захист EUF-CMA необхідно до алгоритму додати довжину «солі» $\overline{\ell}$ до обох ключів: приватного та публічного.

Нехай є документ d, який необхідно підписати. Спочатку необхідно обчислити значення хеш функції $h = H(d) \in \mathbb{F}^m, H : \{0,1\} \to \mathbb{F}^m$. Підпис $z \in \mathbb{F}^n$ документу d обчислюється наступним чином [7]:

1) Обчислюється $x = S^{-1}(h) \in \mathbb{F}^m$;
2) Обчислюється прообраз $y \in \mathbb{F}^n$ від x центрального відображення F;
3) Обчислюється підпис $z = T^{-1}(y) \in \mathbb{F}^n$.

Для того щоб отримати захист EUF-CMA, алгоритм підпису необхідно змінити наступним чином. По-

перше, необхідно випадковим чином згенерувати значення змінних vigenar $\in \mathbb{F}^{v_1}$, після чого вибрати випадкову сіль $r \in \{0,1\}^{\ell}$ та виконати стандартну процедуру підпису Rainbow для $h = H(H(d) \| r)$ щоб отримати підпис $\sigma = (z \| r)$. Якщо лінійна система на кроку 2 генерації підпису не має рішень, то необхідно згенерувати нове значення r та повторити спробу підпису.

Стандартна схема перевірки підпису потребує наступних кроків. Нехай є документ d та підпис $z \in \mathbb{F}^n$. По-перше, необхідно обчислити значення хеш-функції $h = H(d) \in \mathbb{F}^m$, потім обчислити значення публічного відображення у точці z, $h' = P(z) \in \mathbb{F}^m$. Якщо $h' = h$ підпис є коректним. Для того щоб отримати EUF-CMA, необхідно замість $h = H(d) \in \mathbb{F}^m$, обчислити $h = H(H(d) \| r) \in \mathbb{F}^m$.

Серед його сильних сторін можна відзначити малу довжину підпису, а також швидкість створення та перевірки підпису. За рівнем криптостійкості ця схема не поступається жодному з кандидатів, однак довжина особистого ключа є значно більшою, ніж у конкурентів. На противагу цьому швидкість створення ключової пари дуже велика.

V. GeMSS

GeMSS - це багатоваріативна схема підпису, що створює невеликі підписи. Алгоритм має швидкий процес перевірки підпису та середній / великий відкритий ключ. GeMSS будується з криптосистеми прихованих рівнянь поля (HFE), використовуючи так звані модифікатори мінусу та оцту. GeMSS - це більш швидкий варіант QUARTZ, який включає найновіші результати в багатовимірній криптографії для досягнення більш високих рівнів безпеки.

Основним недоліком схеми виступає розмір відкритого ключа, однак автори проекту зазначають, що генерація пари (відритий та секретний ключі) залишаються досить ефективними в GEMSS [4]. Найбільш відомою атакою залишається [5], яка служить орієнтиром для встановлення параметрів для GeMSS.

VI. Порівняння

Порівняння схем, що наводилося в роботі [6] проводилося щодо умовних та безумовних критеріїв. На рисунку 1 відображено гістограму відносної переваги алгоритмів, що базується на цих критеріях.

Умовними критеріями виступили:

1) рівень криптостійкості;
2) довжина відкритого ключа;
3) довжина особистого ключа;
4) довжина результату криптоперетворення;
5) швидкість створення ключової пари;
6) швидкість прямого криптоперетворення;
7) швидкість зворотного криптоперетворення.

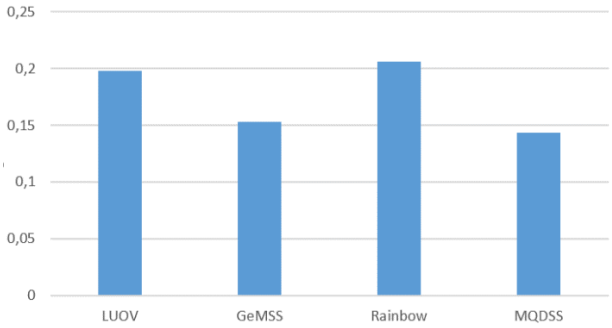


Рис. 1. Відносна перевага алгоритмів ЕП на базі MQ-перетворень

Як видно з рисунка 1, алгоритм Rainbow має найвищий показник, а отже, є найперспективнішим з вищеописаних алгоритмів і допоки кращим кандидатом для використання на міжнародному рівні.

Висновки

Усі схеми, які базуються на MQ перетвореннях попередньо гарантують забезпечення моделі захисту EUF-CMA. Для цих алгоритмів існує добре прорахована математична модель і теорія стосовно забезпечення захисту. Беручи до уваги порівняльний аналіз алгоритмів, маємо те, що найкраще себе показали LUOV та Rainbow. Однак автори схем постійно працюють над ними і вдосконалюють, тож результати можуть змінитись.

Наразі ж найбільш перспективним є Rainbow, позначається плідна багаторічна праця над алгоритмом. Пильної уваги заслуговую також LUOV, однією з особливостей якого є декілька режимів роботи алгоритму верифікації підпису.

Література

[1] LUOV: Signature Scheme proposal for NIST PQC Project.

[2] Стаття Дерека Ціммера на сайті Privacy News Online resource: https://www.privateinternetaccess.com/ blog/2019/02/nist-round-2-and-post-quantum-cryptography-the-new-digital-signature-algorithms/

[3] MQDSS specifications . Version 2.0. Online resource: http://mqdss.org/files/MQDSS_Ver2.pdf

[4] GeMSS: A Great Multivariate Short Signature. Online resource: https://www-polsys.lip6.fr/Links/NIST/ GeMSS_specification.pdf

[5] Jean-Charles Faug`ere and Antoine Joux. Algebraic cryptanalysis of hidden field equation (HFE) cryptosystems using gr¨obner bases. In Dan Boneh, editor, Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings, volume 2729 of Lecture Notes in Computer Science, pages 44–60. Springer, 2003

[6] Applied Radio Electronics. Scientific and Technical Journal. Volume 17 №3,4 – P.138-146.

Дослідження можливості підвищення криптостійкості схеми Dilithium шляхом збільшення розмірності елементів простору ключей

С.О. Кандій[1]

1. Кафедра безпеки інформаційних систем і технологій, Харківський національний університет імені. В. Н. Каразіна, Україна, м. Харків, площа Свободи, 6
sergeykandy@gmail.com

Abstract - The paper considers the possibility of increasing the cryptosecurity of the Dilithium system by increasing the size of the vectors used during transformations. Results are provided for 256,384,512 bits of security.

Key words - DILITHIUM, digital signature, algebraic lattice, parameters, MSIS, MLWE, post quantum.

I. Вступ

Вперше криптографічна схема на основі навчання з помилками була запропонована у роботі ізраїльського криптолога Regev у 2005 році [5]. Побудована одностороння функція з лазівкою мала значні переваги перед існуючими схемами , стійкість яких базується на алгебраїчних решітках, оскільки задача криптоаналізу була in-average-case складною, навідміну від in-worst-case складних проблем, наприклад, як у NTRU. Проблема навчання з помилками полягає у вирішенні системи лінійних рівнянь з адитивним шумом та є NP проблемою. Проте, знаючи адитивний шум можна знайти рішення за поліноміальний час [5]. У загальному випадку систему лінійних рівнянь з адитивним шумом можна записати як $t = A^{k\times l} * s_1^l + s_2^k$, де s_1^l та s_2^k є вектором невідомих та вектором шуму відповідно. Криптоаналіз систем на базі навчання з помилками полягає у знаходженні s_1^l та s_2^k знаючи значення t та $A^{k\times l}$. У сучасних криптосистемах вектори визначені переважно над полем $R_q^n\,[X]/(X^n - X - 1)$ [1,2]. Дослідження можливості збільшення криптостійкості систем на основі навчання з помилками є актуальною задачею, оскільки вони є одними з найбільш перспективних кандидатів для постквановтого періоду. У даній доповіді розглядається можливість підвищення криптостійкості таких систем без суттєвих змін у наборі загальносистемних параметрів.

II. Огляд основних атак

Основні атаки на криптосистему Dilithium зводяться до проблем MLWE та MSIS.

У [1] для оцінки безпеки застосовувалася методика “core SVP hardness”. Вона базується на тому факті, що досі не знайдено ефективних методів експлуатації структури модульних решіток, тож можна звести MLWE та MSIS до LWE та SIS без втрати точності аналізу. Безпеку останніх можна звести до проблеми SVP, яка є гарно вивченою (порівняно з іншими проблемами на решітках). Проте, з кожним роком з’являються нові більш ефективні алгоритми для вирішення цієї проблеми. Найкращим відомим алгоритмом для редукції решіток є BKZ. Він мінімізує кожен вектор решітки у ортогональному підпросторі розмірності b. Найкраща відома складність атаки для класичних комп’ютерів складає $O(2^{0.292b})$ та $O(2^{0.265b})$ для квантових. Під час роботи BKZ викликає “оракула” для мінімізації у ортогональному підпросторі. Найкраща відома складність оракула складає $O(2^{0.2075b})$ [1].

Проблема $MLWE_{k,l,D}$ зводиться до $LWE_{nk,nl,D}$, яку у свою чергу можна звести до $unique - SVP$ на решітці розмірності $d = n * l + n * k + 1$. Норму найменшого вектора можна оцінити як $\lambda \approx \xi * \sqrt{d}$, де ξ є середньоквадратичним відхиленням для розподілу, що використовується. Оскільки в Dilithium використовується рівномірний розподіл, то $\xi = \sqrt{\left(\frac{\sum i^2}{2*\eta+1}\right)}$. Атака вважається успішною, якщо була знайдена довжина блока, для якої проекція найменшого вектора v у векторний простір, натягнутий на останні b векторів Грамма-Шмідта коротше, ніж b_{d-b}. Довжину вектора v можна оцінити як $\delta^{2*b-d-1} * q^{\frac{m}{d}}$, де $\delta = \left(\frac{(\pi*b)^{\frac{1}{b}}*b}{2*pi*e}\right)^{\frac{1}{2(b-1)}}$, а норму проекції як $\xi\sqrt{b} \le \delta^{2*b-d-1} * q^{\frac{m}{d}}$.

Також, іноді застосовується дуальна атака. Атака полягає у вирішенні $decision - LWE$ на дуальній решітці. Припустимо, що було знайдено певний вектор. Якщо рішення належить до LWE, то його коефіцієнти мають мати нормальний розподіл, інакше матиме рівномірний розподіл. Відстань між нормальним та рівномірним розподілом можна оцінити як $\epsilon = 4exp\left(-2 * \pi^2 * \left(\frac{l*\xi}{q}\right)^2\right)$, де l – довжина вектора, ξ – середньоквадратичне відхилення. Це значення можна трактувати як ймовірність успіху. Нажаль вона є досить мала. Для успішної атаки потрібно атаку виконати щонайменше $R = max\left(1, \frac{1}{2^{(0.2075*b)\epsilon^2}}\right)$разів. Довжину вектора можна оцінити як $\delta^{d-1} * q^{\frac{n*l}{d}}$, де d – є розмірність підрешітки, що обрана для атаки.

У [1] було показано, що проблему SelfTargetMSIS можна звести до MSIS. А відповідні екземпляри MSIS можна звести до SIS. Задача SIS полягає у знаходженні вектору цілих чисел, які мають досить малу норму $\| \|_\infty$. Автори [1] пропонують шукати такий вектор за допомогою BKZ (у якому використовується $\| \|_2$ норма). Згідно з [1] , після редукції решітки, що асоційована з криптосистемою, в векторних умовно можна виділити 3 зони (Рисунок 1). У першій зоні коефіцієнти будуть розподілені рівномірно за модулем q. У другій зоні коефіцієнти матимуть нормальний розподіл. У третій зоні усі

коефіцієнти дорівнюють 0. Крипостійкістю є обернена величина до добутку ймовірності того, що усі коефіцієнти менші за певне значення (що є параметром атаки) на час роботи підпроцедури, що виконує редукцію в ортогональному підпросторі в алгоритмі BKZ. Для виявлення меж зон i,j використовується ряд евристичних міркувань [1,2]. До редукції перші вектори матимуть евклідову норму q, а останні 1. Після редукції з'явиться певний "схил". Довжина цього схили і визначатиме межі другої зони. Згідно з [1] , "крутизну схилу" можна обчислити як

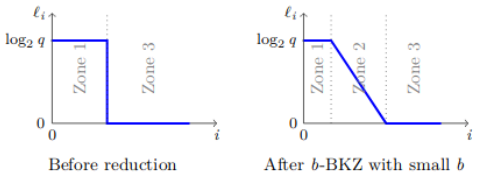
$$slope(b) = \frac{-1}{b-1}\left(\frac{(\pi*b)^{\frac{1}{b}}*b}{2*pi*e}\right)$$

При цьому сума евклідових норм від векторів у ортогональному підпросторі буде тією ж самою, що дозволяє на основі розміру підпростору визначити початок і кінець схилу та у свою чергу оцінити складність атаки.

Зауважимо, що існує ряд теоретичних результатів щодо алгебраїчних атак, але на практиці вони не можуть бути ефективно застосовані, тож при оцінці криптобезпеки їх моделювання не було здійснено.

РИСУНОК 1

РЕДУКЦІЯ БАЗИСУ РЕШІТКИ



III. Результати

Було проведено моделювання розглянутих атак та визначено, що крипостійкість сильно залежить від довжини векторів. Проте ці параметри тісно пов'язані з параметрами поля. Несогласованість може призвести до сильного погіршення швидкодії системи. У Таблиці 1 наведені отримані значення довжини векторів для криптостійкостей 256,384,512 біт відповідно. Для 384 та 512 були замінені параметри поля, оскільки при використанні старих параметрів кількість ітерацій при шифруванні збільшувалася занадто швидко.

ТАБЛИЦЯ 1

ДОВЖИНИ ВЕКТОРІВ ДЛЯ 256,384,512 БІТ БЕЗПЕКИ

	(n,q)	(l,k)
256	(256,8380417)	(9,8)
384	(512,67043329)	(7,6)
512	(512,67043329)	(9,8)

Висновки

1) Для забезпечення криптостійкості сучасних систем на базі MLWE розмір векторів повинен бути близько 10. Це значення можна зменшити, збільшуючи параметри поля, проте на практиці це не доцільно, оскільки час виконання збільшується швидше, ніж при зміні розміру векторів.

2) Невелике збільшення розміру вектора дає великий приріст у криптостійкості. При формуванні загальносистемних параметрів для систем на базі MLWE потрібно враховувати безліч факторів, але в першу чергу потрібно визначити саме розміри векторів та параметри поля, оскільки це є основою для інших параметрів.

3) У криптосистемі Dilithium неможливо досягти збільшення криптостійкості, змінюючи лише розмір векторів. При фіксованих параметрах поля збільшення розміру векторів призведе до занадто великого збільшення кількості ітерацій, що зробить криптосистему непридатною до застосування.

4) Оскільки збільшуючи розмір векторів можна досягати збільшення криптостійкості без зміни поля, то це дає більшу гнучкість при знаходженні компромісу між криптографічною безпекою системи та ресурсами, що потрібні для її реалізації, порівняно з NTRU-подібними системами.

5) Існуючі методики оцінки криптографічної безпеки систем на базі алгебраїчних решіток мають переважно емпіричний характер і зі збільшенням розмірності точність оцінки падає. Необхідність в більш точних методах оцінки криптостійкості зростає все більше з кожним роком.

Література

[1] Lyubachevsky V., Ducas L., Kiltz E. [et all]. CRYSTALS–Delithium. Techn. rep. NIST (2019) / https://crc.nist.gov./projects/post-quantum-cryptogtraphy/ round-2-submissions.

[2] Alkim E., Ducas L., Poeppelmann T., Schwabe P. Post-quantum key exchange – a new hope / http://cryptojedi.org/papers/#newhope, 2016.

[3] Bos J.W., Costello C., Ducas L. [et all]. Frodo: take of the ring! Practical, quantum-secure key exchange from LWE // Proc. of ACM CCS 16, ACM Press, Okt. 2006, P. 1006-1018.

[4] Kiltz E., Lyubachevsky V., Schaffner C. A concrete treatment of Fiat-Naor signatures in the quantum random-oracle model // Cryptology ePrint Archive, Report 2017/916, http://eprint.iacr.org/2017/916.

[5] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. InProc. 37th ACMSymp. on Theory of Computing (STOC), pages 84–93, 2005.

Аналіз стійкості криптосистеми McEliece

Владислав Керничний[1], Олександр Сєвєрінов[1]

1. Кафедра безпеки інформаційних технологій,
Харківський національний університет радіоелектроніки,
УКРАЇНА, г. Харків, пр. Науки, 14,
E-mail: vladyslav.kernychnyi@nure.ua
oleksandr.sievierinov@nure.ua

Коротка анотація – Questions of construction of post-quantum cryptosystems based on algebraic codes are considered. McEliece's robustness to cryptographic attacks is analyzed.

Ключові слова – криптосистема на основі алгебраїчних кодів, криптосистема McEliece, код Goppa, атаки

I. Вступ

На основі сучасних уявлень про квантові комп'ютери існує гіпотеза, що після своєї появи вони будуть здатні зруйнувати більшість, якщо не абсолютно всі традиційні криптосистеми, що широко використовуються на практиці.

На даний момент немає однозначного рішення, щодо вибору кращої постквантової криптосистеми. Існує декілька напрямів, що побудовані на різних алгебраїчних структурах:

– схеми, побудовані на алгебраїчних решітках, мають неясні перспективи зростання рівня безпеки;

– схеми багатовимірної криптографії, побудовані на рішеннях рівнянь, заснованих на багатовимірних поліномах над кінцевим полем, мають такі ж невизначені перспективи криптографічної стійкості для квантового комп'ютера;

– схеми основі хеш-функцій;

– криптосистеми на основі алгебраїчних кодів.

Тому актуальним є дослідження стійкості існуючих криттосистем на основі алгебраїчних кодів, а саме системи McEliece.

II. Криттосистема Mceliece

Перша криптосистема побудована на алгебраїчній теорії кодування була запропонована в 1978 році Робертом МакЕлісом [1]. У запропонованій МакЕлісом початковій конструкції використовуються двійкові коди Goppa [1, 2], але алгоритм може бути використовується з будь-якими лінійними кодами.

Відкритий ключ визначає випадковий бінарний код Гоппа. Зашифрований текст - це кодове слово та випадкові помилки.

Приватний ключ дозволяє досить швидко виконувати наступні дії: витягати кодове слово з зашифрованого тексту, виявляти та видаляти помилки.

Система McEliece була спроектована як одностороння (OW-CPA) - це означає, що зловмисник не може швидко знайти кодове слово з зашифрованого тексту та відкритого ключа, під час випадкової вибірки кодового слова.

На даний час рівень безпеки системи McEliece залишається на досить високому рівні, незважаючи на десятки задокументованих нападів на протязі 40 років. Початкові параметри McEliece були розраховані на 64 біта, але систему легко модифікувати та зробити актуальною при різних ступенях розвитку комп'ютерної техніки, включаючи квантові комп'ютери.

Криптографічна стійкість системи заснована на двох складних обчислювальних завданнях: вичерпному пошуку по ключовому простору і декодуванню по максимуму правдоподібності.

III. Атаки на криптосистему Mceliece

У літературі описана досить велика кількість атак на McEliece [3-5]. Деякі атаки, так звані структурні атаки, засновані на спробі побудувати декодер для коду, згенерованого відкритим ключем G. Якщо така спроба виявиться успішною, то закритий ключ буде розкритий, а криптосистема повністю зламана. Код C*, породжений матрицею G* і код C, породжений матрицею G, належать одному еквівалентному класу. Зловмисник повинен порівняти елементи коду з кожного класу в C* для того, щоб визначити еквівалентний код. Оскільки еквівалентні класи мають дуже малу потужність, цей процес виходить за рамки можливостей навіть найпотужніших комп'ютерів. Для оригінальних параметрів (1024, 524, 50) дана структурна атака вимагає для вивчення більш 2^{466} кодів.

Інші атаки спрямовані на отримання вихідного тексту повідомлення з шифрованого повідомлення. Більшість з них засновані на декодуванні безлічі даних (ISD). Або на парадоксі днів народження, їх узагальненнях і покращеннях.

Існують такі атаки, як, наприклад, ітераційне декодування і статичне декодування, але вони не є успішними. Атака ISD виявилася найменш складною. В останні роки було описано декілька алгоритмів і їх покращенн. Найбільш важливі перераховані у Таблиці 1 разом з їх двійковим показником витрат для декодування (1024, 524, 50) коду Гоппа. Для цих алгоритмів відомі їх граничні показники.

ТАБЛИЦЯ 1

Алгоритми ISD-атак с двійковим показником складності

Рік	Алгоритм	Складність ($\log_2$)
1986	Адамс-Мейер	80,7
1988	Лі-Брікелл	70,89
1989	Штерн	66,21
1994	Кантеаут-Шабант	65,5
1998	Кантеаут-Шабант	64,1
2008	Бернштейн-Ланг-Петерс	60,4
2009	Фінназ-Сендреір	59,9

Найактуальніші атаки використовують інформаційне декодування, або ISD. Найпростіша форма ISD, будується на аналізі інформації про помилки в коді. Інформаційний набір, за визначенням, це набір позицій, який визначає ціле кодове слово. Набір вважається безпомилковим, якщо він уникає всіх помилок в отриманому слові, тобто, в зашифрованому тексті. Зловмисник визначає решту кодового слова за допомогою лінійної алгебри, і перевіряє чи була атака успішною, шляхом перевірки ваги помилки t.

Очікується, що випадковий набір позицій k, буде інформаційним набором з великою вірогідністю. Проте, кількість встановлення безперервно падає зі зростаючою кількістю помилок. Наступне асимптотичне твердження є правильним для будь-якого справжнього числа R в проміжку $0 < R < 1$: якщо розмірність коду $k \in (R + O(1))\times n$, і кількість помилок $t \in \Theta(n/\log_n)$, тоді імовірність того, що набір буде безпомилковим буде дорівнювати $(1 – R + O(1))^e$ та n буде прагнути до нескінченності. Складність ISD є такою $(1/(1\text{-}R) + O(1))^e$ [3-5].

Подальші модернізації ISD вплинули на O(1), але не змінили константу 1/(1-R). В системі McEliece t є асимптотичним $(1 – R + O(1))n/\lg_n$, тому припущення $t \in \Theta(n/\log_n)$ є справедливим. Таким чином рівень безпеки McEliece проти цієї атаки дорівнює $n/\lg_n$ з $1/(1 - R)^{1-R} + O(1)$.

Тим часом розмір шифртексту є $(1 – R + O(1))\times n$ бітів, та розмір ключа дорівнює $(R(1 - R) + O(1))\times n^2$ бітів. Таким чином, рівень безпеки 2^5 використовує ключ розміром $(C_0 + O(1))\times b^2(\lg b)^2$, де $C_0 = R/(1 - R)(\lg(1 - R))^2$. Мінімальне значення, яке може досягти C_0 дорівнює близько 0.7418860694, коли R рівне близько 0.7968121300 [3-6].

Традиційний підхід до атак підібраного шифртексту на систему McEliece – це додавання помилок до зашифрованого тексту Gm + e. Це еквівалентно додаванню помилок до e. Розшифрування виконується тоді і тільки тоді, коли отриманий вектор помилок має вагу t, тобто, точно одна із декількох позицій помилки вже буде в e. Таким чином можливо знайти e.

Але ці напади не спрацовують проти актуальної версії системи McEliece. По-перше, декапсуляція змушує зашифрований текст включати хеш e як підтвердження, а зловмиснику неможливо обчислити хеш модифікованої версії e. По-друге, механізм інкапсуляції ключів не виявляє помилок дешифрування: модифікований шифрований текст буде генерувати непередбачуваний сеансовий ключ, незалежно від того, чи є вбудований вектор помилок вагою t.

Висновки

До сих пір криптосистема McElice з кодами Гоппа не піддається криптоаналізу. Найбільш відомі атаки використовують алгоритм декодування множини даних. В останніх реалізаціях показано як атаки на систему, так і її захист від них. В інших показано, що для квантових обчислень розмір ключа повинен бути збільшений на чотири порядки через удосконалення декодування множини даних. Криптосистема має кілька переваг [7]. Шифрування і дешифрування проходить швидше і з ростом довжини ключа ступінь захисту даних зростає набагато швидше. Довгий час вважалося, що McEliece не може бути використана для ЕЦП. Однак виявилося можливим побудувати схему для ЕЦП (наприклад криптосистема Нідеррайтера).

Недоліком криптосистем McEliece є великий обсяг ключової інформації - від декількох мегабайт. Крім того, вони потребують навіть більших ключів для захисту від квантових комп'ютерів. Через недоліки McEliece використовується рідко. Один з винятків - використання McElice для шифрування в Freenet-подібної мережі ENTROPY.

Література

[1] R. J. McEliece A Public-Key Cryptosystem Based On Algebraic Coding Theory // DSN Progress Report 42-44. — 1978.

[2] Jochemsz E. Goppa Codes & the McEliece Cryptosystem //Vrije Universiteit Amsterdam. – 2002. – С. 63.

[3] Bernstein D. J., Lange T., Peters C. Attacking and defending the McEliece cryptosystem //International Workshop on Post-Quantum Cryptography. – Springer, Berlin, Heidelberg, 2008. – С. 31-46.

[4] Landais G., Tillich J. P. An efficient attack of a McEliece cryptosystem variant based on convolutional codes //International Workshop on Post-Quantum Cryptography. – Springer, Berlin, Heidelberg, 2013. – С. 102-117.

[5] Baldi M. LDPC Codes in the McEliece Cryptosystem: Attacks and Countermeasures //Enhancing Cryptographic Primitives with Techniques from Error Correcting Codes. – 2009. – Т. 23. – С. 160-174.

[6] Vaidya S., Dutta S. A Study of the McEliece PKE. – 2018.

[7] Халимов Г. З., Северинов А. В. Обеспечение безопасности каналов передачи данных на основе помехоустойчивых кодов //Системы управления и связь.–Х.: ХВУ. – 1996. – С. 116-119.

Роль персонала в системе информационной безопасности: правовые инструменты или корпоративная культура?

Максим Кобрин

1. Кафедра социальной информатики,
Харьковский национальный университет радиоэлектроники,
УКРАИНА, г. Харків, пр. Науки, 14,
Юридическая компания «Кобрин и партнеры»
E-mail: kobrin.law@gmail.com

Коротка анотація – In this paper, issues of enhancing information security through the use of such personnel management tools as the formation of a corporate culture and strengthening the motivational climate in the company, as well as the formation of such an element of information security as staff training. Implementation of the proposed methods will reduce the number of information security incidents and increase the competitiveness of the company.

Ключові слова – информационная безопасность; система безопасности; корпоративная культура; управление персоналом; мотивационный климат.

I. Вступ

Персонал компании является частью информационной системы (ИС). Система информационной безопасности (ИБ) компании является подсистемой ИС, таким образом можно сделать вывод о том, что персонал компании является частью системы ИБ компании [1, 2]. Причиной инцидентов в области ИБ часто является «человеческий фактор». В настоящее время, персонал — это ключевой ресурс компании. Важно получить сотрудника, который сможет полностью удовлетворить функциональный запрос нанимателя [3] и избегать работать с сотрудником, который не сможет его удовлетворить.

II. Актуальность

На наш взгляд, корпоративная культура и мотивационный климат в компании также важны для управления ИБ, как и другие факторы, непосредственно связанные с информационными технологиями. Эти инструменты управления могут обеспечить элементы ИБ, которые невозможно или неэффективно обеспечивать другими способами. И наоборот, корпоративная культура и мотивационный климат могут существенно усилить информационную безопасность компании и, соответственно, повысить конкурентоспособность компании. Таким образом, актуальную задачу повышения уровня ИБ компании, предлагается решить посредством интеграции инструментов управления персоналом, таких как, формирование корпоративной культуры и мотивационного климата компании в систему ИБ. Кроме этого, предлагается дополнить систему ИБ подсистемой обучения персонала в области ИБ.

III. Основная часть исследования

Цель любого предприятия в отношении персонала - «безубыточные» для компании отношения. Из распространенных правовых инструментов, с помощью которых обеспечивается отсутствие или минимизация ущерба фирме, применяются такие инструменты как:

1. NDA (Non-disclosure agreement) - соглашение о неразглашении конфиденциальной информации.
2. NCA (Non-competition agreement) - соглашение о неконкуренции.

Однако, эти документы не согласуются с украинским законодательством и не могут в достаточной мере обеспечить те функции, для которых их оформляют (Рис.1).

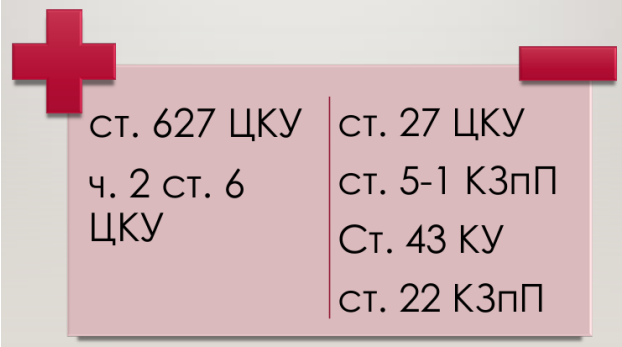


Рис.1. Статьи украинского законодательства, которые применимы к договорам NDA, NCA [4].

Это не значит, что их не нужно оформлять, так как подписанные NDA и NCA могут применяться для дополнительной мотивации сотрудников, как дополнительный психологический фактор обеспечения безопасности.

Роль корпоративной культуры в системе безопасности (СБ) должна быть отражена в политике ИБ, принципах построения и развития компании, в функциях системы безопасности.

Принципы системы безопасности (Рис.2):

1. Законность.
2. Соответствие корпоративной культуре.
3. Своевременность и активность (упреждающий характер мер обеспечения безопасности).
4. Экономическая целесообразность (сопоставимость возможного ущерба и затрат на обеспечение безопасности).
5. Специализация.

Например, некоторые пункты в функциях системы безопасности напрямую связаны с персоналом (п. 1, 2, 7, 8):

– Защита законных интересов компании и ее сотрудников.

– Обучение сотрудников по вопросам безопасности.

– Оперативное реагирование на угрозы безопасности компании.

– Защита коммерческой тайны компании.

– Инженерно-техническая защита зданий, сооружений компании.

– Обеспечение физической охраны собственности компании.

– Защита жизни, здоровья, достоинства сотрудников, прав сотрудников.

– Контроль выполнения сотрудниками правил безопасности.

– Разработка нормативной и методической документации по направлениям безопасности.

Рис. 2. Принципы построения и развития системы безопасности.

При подготовке персонала, с точки зрения ИБ, необходимо:

1. Добиться, чтобы выполнение правил безопасности стало частью корпоративной культуры компании (Рис.3).

Рис. 3. Корпоративная культура, как часть системы обучения правилам ИБ и контроля их выполнения.

2. Обеспечить мотивацию сотрудников на выполнение правил безопасности.

3. Обучить работающих в настоящее время сотрудников.

4. Обучать правилам безопасности сотрудников, принимаемых на работу.

5. Регулярно напоминать сотрудникам о правилах безопасности, проводить тренировки.

6. Организовать обучение руководителей (руководители должны быть подготовлены по более сложной программе, так как им придется обучать сотрудников и контролировать выполнение правил безопасности).

Выводы

На практике данный подход был реализован в ряде торговых компаний, например: ООО «Хот-Велл», ООО «Горизонт», ООО «Атлантик-Гейзер.

Особое внимание было уделено интеграции корпоративной культуры и мотивационного климата компаний с системой ИБ. Все сотрудники этих компаний прошли обучение в области ИБ. Кроме этого, регулярно проводятся занятия по ИБ, которые позволяют сотрудникам актуализировать свои знания. Внедрение данного подхода к управлению персоналом позволило существенно повысить информационную безопасность, эффективность и конкурентоспособность компаний. Количество инцидентов в области информационной безопасности снизилось и, в случаях их возникновения, благодаря подготовке персонала, действия сотрудников были эффективными и слаженными, что позволило минимизировать ущерб для компании.

Дальнейшее изучении этого подхода состоит в разработке соответствующих моделей и более глубокой интеграцией с бизнес-процессами компаний.

Литература

[1] Юдін О.К., Корченко О.Г., Конахович Г.Ф. Захист інформації в мережах передачі даних. — К.: Вид-во ТОВ «НВП» ІНТЕРСЕРВІС», 2009. — 716 с.

[2] Грайворонський, М.В. Безпека інформаційно-комунікативних систем / М.В. Гайворонський, О.М. Новиков. – BHV, 2009. – 608 с.

[3] Кобрин М.В. К вопросу об обеспечении эффективности и безопасности индивидуальной информационной системы пользователя / М.В. Кобрин // 22-й Международный молодежный форум «Радиоэлектроника и молодежь в XXI столетии». Зб. Материалов форума. Т.9. – Харьков: ХНУРЭ. 2018. – С.85-86.

[4] NON-COMPETE В УКРАЇНІ. Гаркуша Андрій [Электронный ресурс]. – Режим доступа: www/URL: https://uba.ua/documents/NON-COMPETE В УКРАЇНІ. Гаркуша Андрій.pdf

Порівняльний аналіз перспективних електронних підписів на основі алгебраїчних решіток для пост квантового періоду.

Марія Левіна[1], Надія Писаренко[2]

1. Студентка Харківського Національного університета імені В. Н. Каразіна, факультет комп'ютерних наук.

2. Студентка Харківського Національного університета імені В. Н. Каразіна, факультет комп'ютерних наук.

Коротка анотація - Розгляд сутності та властивостей розвинених ЕС на основі алгебраїчних решіток та порівняльного аналізу механізмів підпису за умовними та безумовними критеріями.

Ключові слова – CRYSTHAL-DILITHIUM, FALCON, qTESLA, електронний підпис, порівняльний аналіз.

I. Вступ

На сьогоднішній день вирішується серйозна задача у створенні постквантових криптографічних методів цифрового підпису.

Також зараз продовжується конкурс NIST США і перший етап цього конкурсу вже був завершений.

На другому етапі залишилось 9 алгоритмів електронного підпису із 17 алгоритмів, що брали участь під час першого етапу: CRYSTALS-DILITHIUM, FALCON, qTESLA, GeMSS, LUOV, MQDSS, Picnic, Rainbow та SPHINCS+.

Метою цієї доповіді є порівняння трьох алгоритмів які базуються на алгебраїчних решітках, що вийшли до другого етапу конкурсу та розгляд основних компонентів цих підписів.

Як і у минулих змаганнях AES та SHA-3, безпека є найважливішим фактором при оцінці постквантових алгоритмів. NIST має намір стандартизувати постквантові алгоритми відкритого ключа для використання в найрізноманітніших протоколах, таких як безпека транспортного шару (TLS), захищена оболонка (SSH), обмін ключами в Інтернеті (IKE), безпека протоколу Інтернету (IPsec) та Розширення системної безпеки доменного імені (DNSSEC). Схеми оцінюються за безпекою, яку вони забезпечують у програмах.

Під час приймання заявок для конкурсу, подавачів заохочували, але не вимагали надання доказів безпеки у відповідних моделях. Для схем шифрування загального використання та встановлення ключів, FRN-Dec16 попросив схеми "семантично захищеного" щодо адаптивного обраного шифротекстового нападу (безпека IND-CCA2). У випадках ефемірного використання NIST буде враховувати семантичну захищеність щодо обраної атаки простого тексту (безпека IND-CPA). Схеми цифрового підпису повинні включати екзистенційно-непідробні підписи щодо адаптивно-обраної атаки повідомлення (безпека EUF-CMA).

Визнаючи, що існують суттєві невизначеності в оцінці місць безпеки алгоритмів постквантового кандидата, NIST визначив п'ять категорій безпеки, щоб можна було краще порівняти рівень безпеки, що надається в публікаціях. Відправникам було запропоновано надати попередню класифікацію відповідно до визначень, наданих у FRN-Dec16, з акцентом на задоволенні вимог до категорій безпеки, продуктивності та / або алгоритм та реалізації.

Було вирішено порівняти ці три підписи за трьома параметрами: 1) безпека/стійкість, 2) технічні вимоги до експлуатації та 3) захищеність від атак. За схожими критеріями проводили вибір кандидатів для другого раунду в конкурсі NIST США.

I. CRYSTALS-DILITHIUM

Безпечність|Стійкість: Захист схеми цифрового підпису Dilithium ґрунтується на складності пошуку коротких векторів у решітках та сама конструкція Dilithium базується на методі "Fiat-Shamir з перериваннями", який використовує вибірку відхилення, для того, щоб зробити схеми Fiat-Shamir на основі решіток компактними та безпечними.

Основна новизна алгоритму Crystal-Dilithium за думкою авторів полягає у тому, що розмір відкритого ключа скорочується в 2,5 рази за рахунок збільшення підпису приблизно на 150 байт. Також для зменшення розміру відкритого ключа використовується алгоритми, які витягують біти елементів "високого порядку" та "нижчого" порядку в Z_q.

Техніко-експлуатаційні вимоги: Конструкція Dilithium базується на методі "Fiat-Shamir з перериваннями" Lyubashevsky, який використовує вибірку відхилення, для того, щоб зробити схеми Fiat-Shamir на основі решіток компактними та безпечними.

Dilithium пропонує досить високу продуктивність і порівняно простий для реалізації.

Захищеність від атак: Найкращі атаки пов'язані з пошуком коротких векторів у деяких решітках. Для атак використовують два підходи: MLWE та MSIS. Основна відмінність між проблемами MLWE та MSIS полягає в тому, що проблема MLWE передбачає пошук короткого вектора у решітці, в якій існує "надзвичайно короткий" вектор. Проблема MSIS, з іншого боку, передбачає просто пошук короткого вектора в випадковій решітці. У ретроспективній термінології проблема MLWE (розшифрувати і чим відрізняється) – це ретроспектива низької щільності, а MSIS – екземпляр ретроспективи високої щільності.

Хоча проблеми MLWE та MSIS визначаються над поліноміальним кільцями, в даний час немає жодного способу використовувати цю кільцеву структуру, тому кращі атаки встановлюються, просто переглядаючи ці проблеми як проблеми LWE та SIS. Проблеми LWE та SIS точно такі самі, як у визначеннях MLWE та MSIS.

Найбільш відомим алгоритмом пошуку дуже коротких ненульових векторів в евклідових решітках є алгоритм Block–Korkine–Zolotarev (BKZ) , запропонований Schnorr та Euchner у 1991 році. Зовсім нещодавно було доведено, що він швидко зближується зі своїм виправленням і вдосконалюється на практиці. Проте, те, що він досягає асимптотично, залишається беззаперечним.

Також можна зауважити, що існують інші атаки. Але для параметрів сімейства алгоритмів BKW та Arora–Ge далеко не конкурентоздатні.

Серед них є:

– алгебраїчні атаки
– щільні атаки на підрешітки
– спеціалізована атака проти ℓ_∞-SIS.

II. FALCON

Безпечність|Стійкість: Істинна вибірка Гауса використовується всередині, що гарантує незначний витік інформації про секретний ключі до практично нескінченного числа підписів (понад 264). Falcon пропонує дуже хороші показники продуктивність. Основна новинка - це дуже швидкий рекурсивний алгоритм відбору проб Гаусса, використовуючи структуру даних про дерево ("дерево сокола").

Техніко-експлуатаційні вимоги: Ця структура вимагає двох складових: Клас криптографічних решіток. Обрано клас решіток NTRU.

Захищеність від атак: Зразок лазівки. Покладаються на нову техніку, яка називаються швидкою вибіркою Фур'є. Коротко, схема підпису Falcon може бути описана таким чином: Falcon = GPV рамка + решітки NTRU + Швидка вибірка Фур'є. Потрібно більше роботи, щоб алгоритм був захищений від атак бічних каналів.

Найвідоміші атаки проти Falcon ґрунтуються на скороченні решітчастої бази, без істотного використання спеціальної структури решітки NTRU. Потрібно більше роботи, щоб алгоритм підпису був захищений від атак бічних каналів.

III. qTESLA

Безпечність|Стійкість: Схема підписів є доказово EUF-CMA-безпечною: надається зменшення безпеки від складності проблеми кільцевого навчання з помилками до EUF-CMA-безпечної схеми. Зменшення (рівня) безпеки надано у квантовій випадковій моделі оракула, тобто квантовому зловмиснику дозволено звернутися до випадкового оракула в суперпозиції. Зменшення безпеки ґрунтується на варіанті схеми над стандартними решітками. Для приведення зменшення, використовується евристичний аргумент. Зменшення безпеки є явним, тобто можна явно визначити зв'язок між ймовірностями успіху вирішення проблеми R-LWE та підробки підписів qTESLA. Зменшення безпеки є суттєвим(щільним), що є бажаною властивістю, тому що при виборі параметрів схеми відповідно до зменшень безпеки, суттєве(щільне) скорочення призводить до менших параметрів і, отже, до підвищення продуктивності. На відміну від інших альтернатив, qTESLA є консервативною, але ефективною схемою підпису, що було показано відповідно до наданого зменшення безпеки. Тобто, екземпляри qTESLA достовірно захищені в (квантовій) випадковій моделі оракула.

Техніко-експлуатаційні вимоги: Ця схема супроводжується несуттєвим скороченням випадкової моделі оракула та жорстким зменшенням квантової випадкової моделі оракула з R-LWE.

Захищеність від атак: Було помічено, що помилка в доказі безпеки вимагає коригування параметрів (що знижує ефективність схеми).

Однією з переваг qTESLA є те, що Гаусова вибірка вимагається лише під час генерації ключів до зразку *s* та *e*. Тим не менш, для деяких додатків може знадобитися ефективна і безпечна реалізація генерації ключів і, зокрема, захищеність від атак часу та кешу.

Висновки

На нинішньому етапі дослідження постквантових криптографічних примітивів можна стверджувати тільки про певні переваги, недоліки та можливості прийняття їх в якості постквантових стандартів. При подальших дослідженнях можуть бути виявлені вразливості щодо атак, які можуть здійснюватись як при застосуванні класичних комп’ютерів, так і квантових.

На наш погляд найбільш опрацьованим і безпечним є CRYSTALS-DILITHIUM з його скороченим розміром ключа в 2,5 рази.

Falcon хороший тим, що справжня вибірка Гауса гарантує мінімальний витік інформації. Але алгоритм все ще вразливий для атак з боку бічних каналів, зі скороченням решітчастой бази.

Основною перевагою методу qTesla є те, що Гаусовим вибірка тільки під час генерації ключів, що робить алгоритм простіше і допомагає уникнути помилок. Але було помічено, що помилка в доказі безпеки потребує корегування параметрів (що знижує ефективність схеми).

Література

[1] Lili Chen, Stephen Jordan, Yi-Kai- Liu, Dustin Moody, Rene Peralta, Ray Perlner, Daniel Smith-Tone. Report on Post – Quatum Cryptography. NISTIR 8105 (DRAFT).

Кіберзагрози, пов'язані з використанням іноземного програмного забезпечення, і досвід окремих держав з їх попередження

Ігор Логінов

Ситуаційний центр забезпечення кібербезпеки Служби безпеки України, УКРАЇНА, м.Київ,
вул. Володимирська, 33,
E-mail dkib@ssu.gov.ua

The article contains characteristics of main threats, what are connected with the foreign software using, tactic of their realization, US and Russian experience of their prevention, and suggestions for Ukraine, what are based on the foreign experience of prevention cyberthreats.

Key words: cyberthreats, foreign software, foreign experience, backdoors, cyberthreats prevention

I. Вступ

Ще у 2016 р. рішенням РНБО України, введеним в дію Указом Президента від 13.02.2017 № 32, Адміністрації Держспецзв'язку та СБУ доручалось розробити комплекс заходів щодо обмеження використання на об'єктах критичної інфраструктури програмного забезпечення, розробленого та/або виготовленого суб'єктами господарювання держави-агресора.

На виконання цього доручення згідно з Указом Президента України № 133/2017 запроваджено санкції до деяких російських розробників програмного забезпечення, що викликало хвилю непорозуміння з боку його численних користувачів в Україні.

Слід зазначити, що внаслідок дефіциту часу вжиті заходи не отримали серйозного аналітичного обґрунтування та інформаційного забезпечення.

З огляду на викладене оприлюднимо деякі результати вивчення цього питання.

II. Основні кіберзагрози, пов'язані з використанням іноземного програмного забезпечення, та тактика їх реалізації

Органами безпеки і оборони провідних країн світу одностайно поділяється думка про перетворення кіберпростору на окрему сферу міждержавного протиборства. Інструментом протиборства у кіберпросторі слугує кіберзброя, – сукупність спеціально організованої інформації, інформаційних технологій, методів і засобів її застосування [1].

В умовах формального миру протистояння у кіберпросторі відбувається у вигляді розвідувально-підривних акцій. Вони проводяться конспіративно за допомогою кіберзброї, яка позбавляється ознак, що демаскують її державну належність, шпигунське або деструктивне призначення. Така кіберзброя маскується під загальнодоступні програмні засоби.

Кіберзброя розвідувально-підривного характеру зазвичай становить собою алгоритмічні і програмні комп'ютерні закладки [2], які на замовлення спецслужб конспіративно впроваджуються у програмне забезпечення на етапі його розробки. Наприклад, давно відомо про співпрацю кіберрозвідки США з потужними розробниками програмного забезпечення Microsoft, Skype, Netscape, Lotus тощо [3]. Оприлюднена версія і про те, що в ході операції "Буря в пустелі" програмною закладкою виведено з ладу систему зв'язку протиповітряної оборони Іраку [4]. В Інтернеті та спеціальній літературі можна знайти численні приклади використання програмних закладок у протиправних цілях особами, непричетними до спецслужб іноземних держав.

При цьому слід враховувати висновки фахівців про надзвичайну складність виявлення програмних і апаратних закладок у процесі верифікації програмного забезпечення[5].

Очевидно, що кібератаки розвідувально-підривного характеру проводяться стосовно комп'ютерних систем об'єктів критичної інфраструктури. Під час їх підготовки на одне з перших місць виходить проблема доставки кіберзброї на об'єкт нападу. Так, до ізольованої від Інтернету комп'ютерної системи впровадити кіберзброю може агент спецслужби. Прикладом цьому впровадження агентом ізраїльської розвідки вірусу Stuxnet в систему управління гірничо-збагачувальним комбінатом у Бушері [6]. Однак, в силу низки причин цей метод зручний для проведення разових розвідувальних чи підривних акцій, і не дуже придатний для масових кібератак.

Для підготовки ж масових кібератак більш раціональним вбачається відшукати спільне для всіх об'єктів нападу програмне забезпечення, яким інформація обробляється у комп'ютерах, підключених до Інтернету. Це дає змогу вирішити декілька підготовчих завдань, зокрема:

- забезпечити доставку кіберзброї в ІТС об'єктів нападу незалежно від їх відомчої належності та функціонального призначення;
- позбавитися необхідності придбання на об'єкті агентури, оскільки кіберзброя доставлятиметься через Інтернет;

- застосувати універсальну кіберзброю, адаптовану до одного зразка програмного забезпечення, поширеного на всіх об'єктах нападу.

У цьому контексті звертаємо увагу на потребу будь-якого об'єкта критичної інфраструктури в автоматизації документообігу, фінансового, кадрового та матеріально-технічного забезпечення, антивірусному захисті тощо.

Тобто, найбільш приданими для масових кібератак на об'єкти критичної інфраструктури вбачається оновлюване через Інтернет програмне забезпечення відкритого документообігу, антивірусного захисту, фінансової звітності і бухгалтерського обліку. Чим більше воно поширене у державі, тим більш масштабним буде деструктивний ефект від кібератаки.

III. Досвід США і РФ з попередження кіберзагроз, пов'язаних з використанням іноземного програмного забезпечення

На тлі звинувачень російських спецслужб у кібервтручанні у президентські вибори, кіберзагрозами, пов'язаними з використанням іноземного програмного забезпечення, занепокоїлись США. Їх увагу привернуло масове використання урядовими структурами Сполучених Штатів антивірусного пакету "Лабораторії Касперського". Але у цій державі доволі давно було запроваджено 8-ступеневу процедуру реєстрації постачальників товарів і послуг для урядових організацій. Свого часу "Лабораторія Касперського" успішно її пройшла, і була внесена до відповідного реєстру. Проте, після виступів кількох конгресменів і членів уряду Міністерство внутрішньої безпеки США видало Оперативну директиву 17-01, якою зобов'язала федеральні установи та організації протягом 90 діб видалити програмні продукти "Лабораторії Касперського" з їх комп'ютерів. Надалі прийнято Закон про забезпечення національної оборони США на 2018 фінансовий рік, окремим положенням якого заборонене використання продуктів та сервісів "Лабораторії Касперського" в урядових структурах. Керуючись законом, відповідальна за підбір і перевірку урядових постачальників Адміністрація загальних служб США виключила "Лабораторію Касперського" з відповідного реєстру, а Міністерство торгівлі доповнило Регламент федеральної системи закупівель пунктом, яким заборонено укладати з "Лабораторією Касперського" контракти на устаткування, програмне забезпечення та послуги.

У Російській Федерації обрали інший спосіб витіснення іноземного програмного забезпечення з урядових організацій, а саме, програму імпортозаміщення. У сфері обігу програмного забезпечення було сформовано правову основу імпортозаміщення, яку утворено:

Федеральним законом № 188-ФЗ "Про внесення змін до Федерального закону "Про інформацію, інформаційні технології та захист інформації" та статтею 14 Федерального закону "Про контрактну систему у сфері закупівель товарів, робіт, послуг для забезпечення державних і муніципальних потреб";

Постановою Уряду Російської Федерації від 16.11.2015 р. № 1236 "Про встановлення заборони на допуск програмного забезпечення іноземного походження з метою здійснення закупівель для забезпечення державних і муніципальних потреб";

Постановою Уряду РФ від 4.08.2015 р. № 785 "Про створення урядової комісії з імпортозаміщення".

На підставі цих нормативно-правових актів вжито наступні заходи:

Міністерством цифрового розвитку, зв'язку та масових комунікацій РФ розробляються річні плани імпортозаміщення програмного забезпечення;

створено Єдиний реєстр російських програм для електронних обчислювальних машин і баз даних, постановою Уряду від 16.11.2015 р. № 1236 затверджено правила його формування. Рішення про включення або виключення з реєстру приймаються на основі висновку експертної ради представників державних замовників та російської ІТ-галузі;

згідно з постановою Уряду РФ від 20.11.2017 р. № 1594 організовано Єдиний реєстр програм для електронних обчислювальних машин і баз даних з держав – членів Євразійського економічного союзу, за винятком Російської Федерації (т. зв. "Реєстр євразійського програмного забезпечення");

розроблено Методичні рекомендації з переходу державних компаній на переважне використання вітчизняного програмного забезпечення, затверджені наказом Міністерства цифрового розвитку, зв'язку та масових комунікацій РФ від 20.09.2018 р. № 486 тощо.

Крім того, ще у 2010 р. Розпорядженням Уряду Російської Федерації № 2299-р було затверджено План переходу федеральних органів виконавчої влади і федеральних бюджетних установ на використання вільного програмного забезпечення, згідно з яким до кінця 2014 р. планувалось впровадити пакети базового вільного програмного забезпечення, перенести до них дані з попередніх інформаційних систем та перейти до зберігання даних у відкритих форматах. Зазначимо, що з 2011 р. відкритий формат для офісних документів ОБР на базі міжнародного стандарту ISO/IEC 26300 імплементовано у систему держстандартів Російської Федерації.

Для науково-обгрунтованого вибору програмних альтернатив у Російській Федерації також проведено детальний аналіз програмного забезпечення з відкритим кодом. Результати досліджень викладено в аналітичних документах “Аналіз і порівняльна оцінка різних варіантів створення та експлуатації національної програмної платформи” [7] і “Фінансово-економічне обґрунтування створення національної програмної платформи”[8].

Висновки

Підбиваючи підсумки зазначимо, що на увагу заслуговує досвід обох країн. Виходячи з цього вбачається, що мінімізації кіберзагроз, обумовлених надмірним поширенням програмного забезпечення України, може посприяти комплекс організаційних та правових заходів, спрямованих на:

- запровадження процедур перевірки постачальників програмного забезпечення для органів державної влади та управління, об’єктів критичної інфраструктури найважливіших категорій, створення реєстру перевірених постачальників, розробку порядку включення/виключення постачальників із зазначеного реєстру;
- розробку національної програми поетапного імпортозаміщення у сфері інформатизації;
- розробку за результатами вивчення зарубіжного досвіду пропозицій щодо створення нормативно-правової бази імпортозаміщення;
- активізацію реалізації державних програм щодо переведення органів державної влади та управління на вільне програмне забезпечення, створення національного репозиторію такого програмного забезпечення;
- розробку комплексу заходів з підтримки національних виробників програмного забезпечення.

Література

[1] Концептуальные взгляды на деятельность Вооруженных Сил Российской Федерации в информационном пространстве [Електронний ресурс] // Веб-сайт Міністерства оборони РФ. Режим доступу: http://ens.mil.ru/les/morf/Strategy.doc.

[2] Казарин О. В. Безопасность программного обеспечения компьютерных систем. / [О. В. Казарин]. – М. : МГУЛ, 2003. – 212 с.

[3] Microsoft handed the NSA access to encrypted messages [Електронний ресурс] // Веб-сайт “The Guardian”. Режим доступу: https://www.theguardian.com/world/2013/jul/11/microsoft-nsa-collaboration-user-data

[4] Шабанов И. Программные закладки в бизнес-приложениях [Електронний ресурс]// Веб-сайт “Anti-Malware”. Режим доступу: https://www.anti-malware.ru/software_backdoors

[5] Малюк А. Скрытые угрозы зарубежного программного обеспечения [Електронний ресурс]// Веб-сайт “Мир прогнозов”. Режим доступу:https://www.mirprognozov.ru/prognosis/skritie-ugrozi-zarubejnogo-programmnogo-obespecheniya

[6] Вирус Stuxnet отбросил ядерную программу Ирана на два года назад [Електронний ресурс] / Веб-сайт “Lenta.ru”. Режим доступу : http://lenta.ru/news/2010/12/15/stuxnet/

[7] Обзор результатов проекта по государственному контракту 012/112 от г. (Шифр ИО-04/11) [Електронний ресурс] // Веб-сайт Минкомсвязи России. Режим доступу: https://docplayer.ru/amp/41687598-Obzor-rezultatov-proekta-po-gosudarstvennomu-kontraktu-012-112-ot-g-shifr-io-04-11.html

[8] Финансово-экономическое обоснование создания национальной программной платформы / ООО “ПингВин Софтвер”; руков. П. А. Фролов; исполн.: В. И. Антипов [и др.]. – М.: 2011.–119 с.

Автентифікація користувачів у системах Інтернету речей

Валерія Лукащик[1], Тетяна Гріненко[2]

1. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м. Харків, пр. Науки, 14, E-mail: valeriia.lukashchyk@gmail.com

2. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м. Харків, пр. Науки, 14, E-mail: tetiana.grinenko@nure.ua

Коротка анотація – Thesis is devoted to 'Internet of things' systems, relevance and prospects for their development. Also reviewed information security problems of 'Internet of things' devices and IoT systems. Comparative analysis of user authentication techniques was carried out to further development a method for its evaluation.

Ключові слова – Автентифікація, біометрична автентифікація, Інтернет речей.

I. Вступ

Сучасний етап розвитку суспільства характеризується зростаючою популярністю інформаційних технологій. В даний час відбувається новий виток технічного розвитку цивілізації, який полягає в переході до автоматизації не тільки процесів на виробничих підприємствах, але і процесів, що протікають у повсякденному житті кожної людини. Ідея полягає у використанні великої кількості невеликих малопотужних з обчислювальної та енергетичної точки зору пристроїв для виконання однотипних простих завдань. Така технологія узагальнено має назву Інтернет речей. У сучасному суспільстві Інтернет речей вже став невід'ємною частиною життя багатьох людей. Завдяки появі бездротових мереж, постійного зростання пропускного обсягу Інтернет-з'єднання та впровадження нових підключених пристроїв усе більша кількість людей оточує себе мережевою інфраструктурою, що допомагає і вирішує завдання, які до цього доводилося вирішувати самостійно.

II. Актуальність та перспективи розвитку

Інтернет речей (англ. Internet of things, IoT) - концепція обчислювальної мережі фізичних предметів («речей»), оснащених вбудованими технологіями для взаємодії один з одним або з зовнішнім середовищем, яка розглядає організацію таких мереж як явище, здатне перебудувати економічні та суспільні процеси, що виключає з частини дій і операцій необхідність участі людини [1, 2].

Класифікація Internet of Things за взаємодією між пристроями та :

–BAN (Body Area Network) – людиною, все, що так чи інакше знаходиться на користувачі: розумний годинник, футболки, кросівки, окуляри та інше [3];

–LAN (Local Area Network) – особистою територією: датчики вимірювання різних параметрів, об'єднаних в «розумний дім» [3];

–WAN (Wide Area Network) – міським простором, розумні міста: велосипеди, автомобілі, поїзди, автобуси, літаки, підключені до Інтернету; електростанції, об'єднані в одну мережу і автоматично перерозподіляють навантаження та інше [3];

–VWAN (Very Wide Area Network) – всесвітнім простором, розумна планета, де кожен пристрій може взаємодіяти з будь-яким іншим [3].

На фоні багатьох теорій та моделей розвитку, що жваво обговорюються як на міжнародних, так і на національних форумах, в останні 10-15 років особливу увагу привертає тема впровадження та використання практично у всіх сегментах соціальної активності людства технологій Інтернету речей. Якщо підсумувати численні різноманітні прогнози експертів та фахівців, то можна очікувати у 2025 – 2030 роках наступне [4]: 80-100 мільярдів підключень до мережі Інтернет (сьогодні – біля 16 млрд.); 7-19 трильйонів дол. буде складати світовий ринок IoT; один трлн. євро – ринок технологій IoT у Європі; Індустрія 4.0, як складова Інтернету речей, дозволить отримати додатковий дохід: 30 млрд. євро – Німеччина та 110 млрд. євро – Євросоюз. Прогнозується, що у 2030 році технології Інтернету речей дозволять забезпечити: 10-15 % економії бюджету на охорону здоров'я; 10-15 років збільшення тривалості життя; 40-50 % збільшення врожайності; 15-20 % збільшення пропускної здатності доріг у містах; до 85-90 % зменшення кількості автомобілів; у 10-15 разів зменшення витрат на логістику тощо.

III. Автентифікація користувачів в IoT

Завдання автентифікації – упевнитися, що користувач дійсно має право доступу до пристроїв, додатків та інформації, якою вони оперують.

Для коректної автентифікації користувача необхідно, щоб користувач пред'явив автентифікаційну інформацію – якусь унікальну інформацію, якою повинен володіти тільки він і ніхто інший.

Існує три основних типи автентифікації інформації:

–користувач, що перевіряється, знає якусь унікальну інформацію (автентифікація за допомогою пароля);

–користувач має якийсь предмет з унікальними характеристиками або вмістом (смарт-карта, USB-токен і т.д.);

–автентифікаційна інформація є невід'ємною частиною користувача (відбиток пальця і інші види біометричної автентифікації).

Головним напрямком для вирішення проблем інших методів автентифікації – є вдосконалення методів автентифікації користувачів за рахунок застосування біометричних технологій, які дозволяють забезпечити доступ до інформаційних систем виключно легітимним користувачам, а також обмежити доступ

до інформаційних систем зловмисникам. На даний момент спостерігається тенденція трансформації біометричних технологій в повноцінний компонент систем захисту, тому все більшого поширення набувають системи біометричної автентифікації користувача. Системи біометричної автентифікації – системи, які використовуються для автентифікації особи користувача на основі його біометричних даних [5].

В процесі біометричної автентифікації еталонний і пред'явлений користувачем зразки порівнюються з деякою допустимою похибкою, яка встановлюється заздалегідь. Похибка підбирається для встановлення оптимального співвідношення двох основних показників біометричної автентифікації [6]:

–FRR (False Reject Rate) – коефіцієнт помилкової відмови, також називається помилкою 1-го роду (ймовірність того, що людина може бути не розпізнана системою);

–FAR (False Accept Rate) – коефіцієнт помилкового прийняття, також називається помилкою 2-го роду (ймовірність того, що одна людина може бути прийнятий за іншу).

Обидві характеристики отримують розрахунковим шляхом на основі методів математичної статистики і вимірюються у відсотках. Чим нижче ці показники, тим точніше розпізнавання об'єкта. Для найпопулярніших на сьогоднішній день методів біометричної автентифікації середні значення FAR і FRR виглядають наступним чином (табл. 1).

ТАБЛИЦЯ 1

СЕРЕДНІ ЗНАЧЕННЯ FAR І FRR

Біометричний метод	FAR	FRR
Відбиток пальця	0.001%	0.6%
Геометрія руки	0.3%	4%
Сітчатка ока	0.0001%	0.4%
Рисунок вен	0.0008%	0.1%
Розпізнавання по 2d образу обличчя	0.4%	7%
Райдужна оболонка ока	0.00001%	0.016%

При побудові ефективної біометричної системи контролю доступу недостатньо високих показників FAR і FRR, тому для якісного аналізу також необхідно використовувати й інші показники:

–можливість підробки біометричних параметрів для автентифікації і доступу до системи;

–швидкість автентифікації, а також можливість швидкого безконтактного сканування біометричних параметрів;

–стабільність або незмінність біометричних параметрів в умовах навколишнього середовища та з плином часу;

–вартість реалізації біометричної системи контролю доступу та доступність самих систем на українському ринку.

Висновок

На сьогоднішній день Інтернет речей стрімко входить як у повсякденне життя людей, так і у важку промисловість та критичні системи. Для коректного функціонування локальних і глобальних мереж необхідно мати чіткий і непохитний зв'язок між приладами цих мереж, а також враховувати усі вимоги інформаційної безпеки цих приладів та систем. Адже від правильності функціонування усіх частин IoT інколи залежить людське життя.

Особливу увагу потрібно приділяти додаткам та пристроям, за допомогою яких виконується контроль системи та пристроїв IoT. А саме, автентифікації користувачів у системі. Усе більше пристроїв володіє великими об'ємами інформації, що стосуються користувача, отже постає проблема несанкціонованого доступу, яка може бути вирішена за допомогою впровадження надійних засобів автентифікації.

Необхідно застосовувати ті чи інші технології і засоби автентифікації залежно від рівня загроз і частоти виникнення ймовірних небезпечних подій. При цьому ризики повинні не тільки оцінюватися на етапі проектування, а й відслідковуватися в процесі експлуатації обраних засобів.

Література

[1] Internet of Things [Електронний ресурс] //Gartner IT Glossary – Режим доступу https://www.gartner.com/it-glossary/internet-of-things/ - 30.10.2019

[2] Kevin Ashton. That 'Internet of Things' Thing. In the real world, things matter more than ideas. (англ.). RFID Journal.

[3] Rob Van Kranenburg: What is IoT? [Електронний ресурс] //Counsil - Режим доступу https://www.theinternetofthings.eu/rob-van-kranenburg-what-iot - 30.10.2019

[4] Баранов О.А. Огляд правових проблем Інтернету речей : матеріали науково-практичної конференції ["Інтернет речей: проблеми правового регулювання та впровадження"], (Київ, 24 жовтня 2017 р.) ; упоряд. : В.М. Фурашев, С.Ю. Петряєв. – К. : Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського". – Вид-во "Політехніка", 2017. – 238 с.

[5] Strategy Analytics: Internet of Things [Електронний ресурс] // Strategy Analytics – Режим доступу https://news.strategyanalytics.com/press-release/iot-ecosystem/strategy-analytics-internet-things-now-numbers-22-billion-devices-where - 31.10.2019

[6] Биометрические системы идентификации и аутентификации [Електронний ресурс] – Режим доступу: http://globuss24.ru/doc/ biometriceskie–sistemy–identifikacii–i–autentifikacii – 31.10.2019

Протокол для забезпечення інформаційної безпеки в системі "Розумний будинок"

Олексій Ляшенко, Денис Гольцев

Кафедра безпеки інформаційних технологій, харківський національний університет радіоелектроніки, УКРАЇНА,
м. Харків, пр.. Науки, 14
E-mail: oleksii.liashenko@nure.ua
denys.holtsev@nure.ua

In this paper, we propose an implementation option for a protocol for secure data exchange between smart home network devices. The models necessary to solve the problem of ensuring secure interaction using low-resource cryptography in a smart home network are considered. A software implementation of the proposed protocol was developed using the example of a smart home network.

Ключові слова – розумний будинок, протокол інформаційна безпека, моделі загроз.

I. Вступ

В рамках даної роботи загальна проблема безпеки Інтернету речей була зведена до окремого випадку - безпеки пристроїв розумного будинку. Метою даної роботи є забезпечення безпеки взаємодії між компонентами середовища розумного будинку. Розумний будинок, як підмножина елементів Інтернету речей в роботі визначено у вигляді набору пристроїв, пов'язаних в єдину мережу, метою якої є автоматизація всіляких побутових процесів [1]. У загальному випадку розумний будинок не є повністю автономної мережею - користувач має можливість втручатися в роботу пристроїв, здійснюючи безпосереднє керівництво. В даний момент на ринку представлені такі методи управління агентами розумного будинку [2]:

− автоматизоване, рішення приймається на основі інформації з датчиків;

− за допомогою централізованої панелі управління (всі пристрої - одна панель управління);

− за допомогою пульта (один пристрій - один пульт);

− комбінований.

II. Модель фізичного середовища

Для створення протоколу, що забезпечує належний рівень безпеки взаємодії в рамках мережі, при цьому не навантажує згадану мережу зайвим функціоналом, необхідно сформувати правильне розуміння середовища функціонування і загроз функціонуванню. Таким чином, необхідно сформувати моделі, що забезпечує як адекватне сприйняття фізичної і віртуальної середовища, так і дозволяють зрозуміти, що саме здатне порушити функціонування системи в згаданих середовищах.

Розглянемо поняття фізичної середовища з точки зору пристроїв розумного будинку. Під фізичним середовищем мається на увазі пряме оточення пристроїв, здатних на взаємодію з ним. Модель фізичного середовища, з точки зору безпеки, не відповідає на питання про адекватність застосування агента в тому чи іншому середовищі. При цьому модель оточення повинна враховувати середу поширення сигналу (як джерело загрози) і користувача (авторизованого, як джерела команд і неавторизованого, як джерело загрози). Слід зазначити, що незважаючи на те, що з точки зору розробляється протоколу не існує жодних методів вплинути на загрози з фізичної середовища, модель згаданого середовища необхідна для розуміння особливостей розробки в сфері розумного будинку і налаштування обладнання. Застосування інтерфейсів авторизації, електронних ключів і інструментів біометрії дозволяє істотно знизити ризики, пов'язані із взаємодією на фізичному рівні. Крім того, неадекватне сприйняття фізичної середовища здатне привести до неправильного функціонування обладнання.

Слід зазначити, що взаємодія на фізичному рівні, це коли порушник має прямий доступ до агентам мережі. Дана модель перетинається з моделлю загроз у сфері порушень, пов'язаних з доступом до середовища поширення інформації.

III. Модель експериментальної системи

Для створення повноцінної моделі експериментальної середовища, необхідно визначити кожен з об'єктів згаданого середовища.

Для спрощення сприйняття моделі визначимо поняття часу виконання операції, у вигляді тимчасового проміжку t між надходженням запиту (із зовнішнього середовища або при виконанні коду на обчислювальній машині). Тоді під адекватним часом виконання операції буде матися на увазі відрізок $t < delta\ t$, де $delta\ t$ - час між останнім запитом на виконання операції і новим запитом на виконання операції. Поняття адекватного часу виконання критично важливо для розуміння можливості впровадження додаткових заходів захисту.

Визначимо базовий контролер у вигляді обчислювальної машини, здатної виробляти складні, з точки зору протоколу, операції за адекватний час. Так як розроблений протокол не має сенсу поза фізичного носія (мережі розумного будинку), наділимо базовий контролер можливістю створювати згадану мережу. Тоді, саме базовий контролер відповідає за реєстрацію нових агентів мережі, зберігання інформації про агентів і передачу інформації в шлюз або на пристрій обробки інформації. Крім того, базовий контролер відповідає за виконання функцій безпеки. З точки зору ієрархії системи знаходиться на першому рівні маючи доступ до другого рівня.

Також була розроблена модель, яка необхідна для розробки протоколу в рамках даної роботи, це модель загроз, спираючись на яку можна судити про адекватність застосування тих чи інших рішень безпеки в протоколі.

Основним завданням даної моделі є визначення кінцевого числа загроз, які несуть найбільший ризик з точки зору протоколу взаємодії в мережі, для подальшого пошуку рішень по мінімізації згаданих ризиків від кожної конкретної загрози. В якості зовнішньою загрози розглядаємо загрозу, джерело якої знаходиться за межами мережі і потенційно має доступ до неї через шлюз. Наявність в моделі зовнішніх загроз виправдовує необхідність застосування заходів захисту до шлюзу. Внутрішньою загрозою назвемо загрозу, джерело якої знаходиться всередині мережі.

Слід зазначити, що в подібному вигляді модель є надмірною. Робота з реальної експериментальним середовищем показала неадекватність проведення атак, пов'язаних з підробленими агентами в рамках даних в цій роботі визначень. На Рис. 1 запропонована схема експериментальної системи – носія розробленого протоколу.

Аналогічно можна виключити компрометацію агента - порушення подібного рівня має на увазі прямий доступ до агенту на фізичному рівні та запобігає застосуванням технічних, а не програмних засобів. При цьому застосування подібних середовищ вимагає наявності обчислювальних потужностей до агента, що, з точки зору моделі, перетворює його в зовнішній контролер.

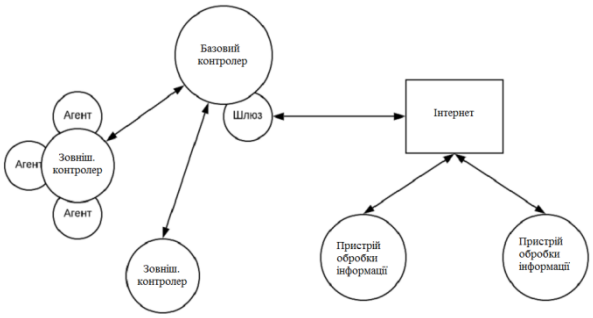


Рисунок 1 – Схема експериментальної системи

IV. Структура базового пакета протоколу

Розглянувши обмеження, що накладаються реалізацією стека протоколів передачі даних, можна зробить висновок, що з будівлі фрейму, що базове обмеження на розмір пакета в реалізованому протоколі становить 2313 біт. Дане обмеження не є критичним, так як підсумковий пакет протоколу залишає істотний запас для передачі. Для надійного механізму підтвердження, необхідно ввести поля флагів і заголовків, для спрощення ідентифікації пакетів і прискорення їх обробки [2].

Так як мова йде про обмін інформацією всередині мережі розумного будинку, необхідність додаткової кодування відсутня - завдання обробки даних лягає на кінцевого адресата даних. Вид отриманого розробленого пакету представлений на Рис.2.

Заголовок	Flag	Дані	Контр. число	Метки
8	8	0 - 263	16	16

Рисунок 2 – Базове представлення пакета

Заголовок включає в себе наступні біти: K - зазначає повідомлення як частина процедури зміни ключа; N - зазначає повідомлення, як частина процедури установки нового пристрою; D - зазначає повідомлення, як частина стандартного обміну даними; M - зазначає повідомлення, як частина процедури вирівнювання міток; BR - зазначає повідомлення, як передачу, яка транслюється на всю доступну підмережа (не скасовує ієрархічні обмеження на обмін даними); B - зазначає повідомлення, як повідомлення від базового контролера; E - зазначає повідомлення, як повідомлення від зовнішнього контролера; A - зазначає повідомлення, як відправлене агентом.

Поле контрольного числа відповідає за підтвердження успішної пересилання. Застосовується спільно з бітами W і E поля прапорів.

Являє собою пару пару $(x, F(x))$, де x - випадкове число, що генерується відправником, а $f(x) = x + Int(add8)\ mod\ 2\ ^\wedge\ 16$ - відповідь. Тут $add8$ - перші 8 інформаційних біт. У разі задіяного прапора C поле даних підлягає розшифруванню перед обробкою.

Висновки

При необхідністі застосування механізмів безпеки всередині мереж розумного будинку, мала потужність елементів подібних мереж накладає істотні обмеження на можливі механізми безпеки. Алгоритми, ефективні на великих масивах даних, при використанні на серверах і персональних комп'ютерах, часто непридатні в умовах відсутності запасу пам'яті і процесорного часу. Таким чином, необхідно застосування більш економічних алгоритмів. Ефективне функціонування мережі досягається так само за рахунок мінімізації витрат на пересилку окремого пакета шляхом скорочення його полів і розміру. При розробці протоколу для мережі, що функціонує на обладнанні малої обчислювальної потужності, необхідно також враховувати особливості обладнання, умови, в яких воно функціонує і призначення подібного обладнання. Так, протокол, оптимальний для мережі розумного будинку буде непридатний для промислового інтернету речей. При виборі криптографічних алгоритмів, що застосовуються в рамках мережі обладнання малої потужності, важливим критерієм є не тільки криптографічний стійкість і обчислювальна простота, а й наявність стандартів застосування даного алгоритму.

Література

[1] Liashenko O. Model of the work of the neurocontroller to control fuzzy data from the sensors of the climate control subsystem “smart house”. / O. Liashenko, O. Barkovska, C. Al-Atroshi, O. Datsok, S. Liashenko // International Journal of Advanced Trends in Computer Science and Engineering, 2019. – №8(1), C. 70-74.

[2] Wenling Wu. LBlock: A Lightweight Block Cipher. / Wu Wenling, Zhang Lei // International Conference on Applied Cryptography and Network Security ACNS 2011: Applied Cryptography and Network Security P. 327-344

Моделювання можливих загроз інформаційної безпеки в системах з використанням мікроконтроллерів AVR

Олексій Ляшенко, Олег Журіло

Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА,
м. Харків, пр.Науки, 14
E-mail: oleksii.liashenko@nure.ua

This paper presents the results of a study of the applicability of elliptic cryptography algorithms on microcontrollers of the AVR architecture. A review of possible threats to the security of information flows for microcontrollers is performed.

Ключові слова – мікроконтролер, інформаційна безпека, моделі загроз, еліптичні криві.

I. Вступ

В рамках роботи проводиться дослідження можливостей реалізації асиметричних криптоалгоритмів високого рівня безпеки на пристроях з дуже обмеженими ресурсами, а саме - реалізація криптографічного модуля на еліптичних кривих для пристрою архітектури AVR.

II. Аналіз захищеності інформаційних потоків при використанні мікроконтролерів AVR

Найбільш цікавими для зловмисників системами з застосуванням AVR є СКУД, системи телеметрії, контролю та обліку, а також автомобільна електроніка. Решта сфери застосування мікроконтролерів даного сімейства також вимагають захисту, але при цьому наслідки їх злому є набагато менш критичними і вигідними, а отже, і менш привабливими для порушників.

Складністю для захисту AVR-пристроїв є їх обмеженість в ресурсах ЦП і доступної пам'яті за сучасними мірками. Навіть прості мобільні телефони давно мають велику обчислювальну потужність. Але навіть на обмеженому просторі можна використовувати такі методи захисту, як шифрування даних, забезпечення автентифікації і забезпечення цілісності даних. Підлаштовуючи апаратну частину для максимальної оптимізації обчислювальних витрат, обсягу пам'яті і споживання енергії, на вільних обчислювальних потужностях можливе використання симетричних (AES, 3DES - підтримуються апаратно в Atmel AVR XMEGA, можлива програмна реалізація, але вона працює на порядок повільніше) і асиметричних криптоалгоритмів (RSA , DH, ECC), обчислення контрольної суми (CRC підтримується апаратно в мікроконтролерах сімейства Atmel AVR XMEGA), використання криптографічних хеш-функцій, комбінація вищевказаних методів.

Основними засобами забезпечення безпеки є програмне або апаратне використання симетричних криптоалгоритмів і криптографічних хеш-функцій, застосування ЕЦП, розвивається використання асиметричних криптоалгоритмів. Atmel для полегшення роботи над захистом даних і оптимізації організації даного захисту, а також для збільшення швидкодії вбудувала в свою провідну лінійку 8-бітних мікроконтролерів AVR XMEGA апаратні реалізації захисних алгоритмів. Користувач при необхідності може вибирати між симетричними алгоритмами шифрування AES і 3DES, а також між контрольними сумами CRC-16 і CRC-32.

III. Застосування еліптичної криптографії на AVR

Асиметрична криптографія незамінна при вирішенні завдання забезпечення шифрування в ситуації, коли між сторонами відсутній взаємна довіра при прийомі інформації.

Одними з найбільш швидких, економічних і прогресивних асиметричних криптоалгоритмів вважаються алгоритми, засновані на еліптичній криптографії [1].

Характеристики алгоритмів еліптичної криптографії можна представити наступним чином. Еліптична крива – це безліч точок (x, y), що описуються рівнянням: $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + +a_6$.

Дане рівняння може розглядатися над довільними полями, і кінцевими полями в тому числі. В такому випадку рішенням рівняння буде безліч окремих точок, а не лінія, що представляє для криптографії інтерес, у зв'язку з можливістю програмної реалізації.

Для використання еліптичних криптоалгоритмів повинні бути узгоджені параметри, що визначають еліптичну криву - набір параметрів протоколу.

Крипостійкість всіх заснованих на еліптичних кривих алгоритмів забезпечується на відсутності субекспоненціальності алгоритму розв'язання задачі дискретного логарифмування в групах їх точок. Складність рішення також визначається порядком групи точок еліптичної кривої.

Існує досить велика кількість алгоритмів еліптичної криптографії, але далеко не всі з них є популярними. Найбільш часто вживаними в захищених системах алгоритмами еліптичної криптографії можна назвати: протокол отримання секретного ключа ECDH, алгоритм ECIES, алгоритм для створення цифрового підпису ECDSA, алгоритм для створення цифрового підпису ECSS.

IV. Моделі загроз та шляхи протидії

Основними загрозами безпеці переданої з мікроконтролера та на нього інформації є [2]:

– перехоплення переданих даних зловмисником;

− аналіз інформації, що передається;
− зміна інформації, що передається;
− глушіння потоку інформації.

Глушіння потоку інформації, а також її перехоплення є загрозами, які перебувають на найнижчому, фізичному, рівні мережевий моделі. Також на даному рівні можна розглянути таку загрозу, як отримання зловмисником фізичного доступу до мікроконтролеру.

Криптографічними алгоритмами можна забезпечити тільки частковий захист від другої загрози, це можливо організувати шляхом спеціального розподілу ключів. Головним в даному випадку дією, спрямованою на забезпечення інформаційної безпеки, буде організація фізичного захисту, яка не дозволяє порушнику дістатися до пристрою.

Для захисту ж від зловмисника з потоком інформації слід налаштовувати фізичні параметри бездротового приймача-передавача даних або організовувати фізичний захист для дротового каналу зв'язку.

Решта загроз розташовуються на мережевому рівні мережевий моделі, і криптографічні алгоритми стають основним засобом захисту та протидії їм. Автентифікація і схеми розподілу ключів дозволяють обмежити в доступі до вузлів мережі несанкціонованих особистостей, які є зловмисниками, доступ отримують тільки перевірені користувачі. Другим же засобом інформаційного захисту є шифрування. Шифрування дозволяє зберегти в секреті передаються по каналах дані в тому випадку, якщо порушник має доступ до каналів.

V. Вибір базового ядра AVR

В силу того, що архітектура AVR є досить складною структурою, а також зважаючи на наявність вільно розповсюджених реалізацій ядер AVR для ПЛІС, було прийнято рішення використовувати сторонню реалізацію, виконану на мові опису апаратури VHDL. Таких було проаналізовано три:

− AVR Core - найбільш відома і часто використовувана реалізація. Являє собою Мікроконтроллерні ядро, сумісне з Atmel ATmega103, має такі ж набір і таймінг інструкцій, як це було зроблено мікроконтролер, прийнятий за зразок для моделювання. Дане ядро розглядалося в якості основного варіанту для запуску на ПЛІС, проте було виведено з роботи.;

− AVRTurnin - дана реалізація могла бути залишена в роботі як другорядна в силу своєї малої обчислювальної здатності, однак для роботи з криптографічними алгоритмами вона має занадто малими можливостями зберігання даних - всього 16 КБ постійної пам'яті і 1кб оперативної;

− pAVR - проект, який реалізує AVR-сумісний мікроконтролер, який не має конкретного зразка для моделювання. Метою реалізації було створення максимально потужною реалізації AVR. Побудований за даною технологією процесор приблизно в 3 рази швидше оригінального ядра.

VI. Результати використання еліптичної криптографії на мікроконтролерах

Результати дослідження показали, що використання еліптичної криптографії на мікроконтролерах архітектури AVR, можливо і піддається реалізації, незважаючи на велику обчислювальну складність еліптичних алгоритмів. Час на виконання криптографічних операцій, на жаль, дуже великий, тому звернення до реальних AVR-мікроконтролерів, що використовують алгоритми еліптичної криптографії, має проводитися не частіше, ніж раз в 30 секунд. У разі застосування софт-реалізацій AVR на основі ПЛІС, що володіють можливістю збільшення максимальної тактової частоти ЦП більш ніж в 3 рази в порівнянні з оригіналом, даний період часу може бути скорочений к співвідношенню використовуваних частот ЦП.

За результатами вимірювань можна зробити висновок, що використання еліптичної криптографії на AVR зажадає близько 39 КБ пам'яті програм і 5,5 КБ пам'яті даних. Вільними на мікроконтролерах сімейства ATmega128 залишаться 89 КБ пам'яті програм і 2,5 КБ пам'яті даних, що дозволяє організувати виконання додаткових операцій на мікроконтролері паралельно з використанням еліптичної криптографії.

Висновки

У роботі було проведено моделювання роботи мікроконтролера, були реалізовані алгоритми шифрування і ЕЦП високого рівня безпеки, засновані на ідеях еліптичної криптографії, а також проведені заміри продуктивності алгоритмів і використання ними доступної пам'яті.

Як показали порівняльні вимірювання з асиметричними алгоритмами і симетричними алгоритмами, використання алгоритмів еліптичної криптографії для підвищення рівня інформаційної безпеки на мікроконтролерах архітектури AVR можливо, не дивлячись на високу обчислювальну складність, але тільки в складі систем, які не потребують передачі даних частіше ніж один раз на 30 секунд. Застосування традиційних асиметричних алгоритмів шифрування на подібних мікроконтролерах можливо тільки при використанні рідкісних флагманських моделей з 32 КБ пам'яті даних через високе її споживання цими алгоритмами.

Література

[1] Горбенко І.Д. Прикладна криптологія. Теорія. Практика. Застосування. / Горбенко І. Д., Горбенко Ю. І. // ХНУРЕ, Приват. акціонер. т-во "Ін-т інформ. технологій". - Х. : Форт, 2013. - 878 с.

[2] Ruchika Markan. Literature Survey on Elliptic Curve Encryption Techniques / Markan Ruchika, Kaur Gurvinder // International Journal of Advanced Research in Computer Science and Software Engineering. 2013. – № 3. С. 906-909.

Сравнение методов стеганографии в изображениях

Виталий Мартовицкий[1], Елизавета Кортяк[1]

1. Кафедра электронных вычислительных машин, Харьковский национальный университет радиоэлектроники, УКРАИНА, г. Харьков, пр. Науки, 14,
E-mail: vitalii.martovytskyi@nure.ua

1. Кафедра электронных вычислительных машин, Харьковский национальный университет радиоэлектроники, УКРАИНА, г. Харьков, пр. Науки, 14,
E-mail: yelyzaveta.kortiak@gmail.com

Steganography is a method of hiding information in other information of different format (container). There are many steganography techniques with various types of container. In the Internet, digital images are the most popular and frequently used containers. We consider main image steganography techniques and their advantages and disadvantages. We also identify the requirements of a good steganography algorithm and compare various such algorithms.

Ключевые слова – стеганорграфия, искажение данных, скрытый канал связи.

I. Введение

Назначение компьютерной безопасности состоит в защите информации от несанкционированного доступа, случайного или целенаправленного искажения данных без изменения основных свойств файлов. Криптография создавалась как методика для защиты систем связи методами кодирования и последующей расшифровки данных. Стеганография дополняет криптографию, скрывая сам факт наличия сообщения в передаваемом потоке данных. Стеганографию можно рассматривать как создание скрытого канала связи.

II. Стеганографические изображения

Скрытое сообщение можно инкапсулировать практически во все виды данных. Большинство инструментов стеганографии ориентируется на передачу сообщения в Интернете, где значительная часть информации передается в виде изображений. При обработке изображения- контейнера учитывают формат файла, в частности методы сжатия. От этого зависят как методы инкапсуляции, так и объем стеганограммы, которую можно вставить в файл. Сложность процедур стеганографии также зависит от формата контейнера.

III. Классификация методов стеганографии для изображений-контейнеров

В течение всего периода развития стеганографии было разработано множество методов, зависящих от форматов изображений и от применяемого аппаратного обеспечения. Методы стеганографии для изображений можно разделить на два класса: методы для временной области [1, 2] и методы для частотной области [1–3]. Для временной области основные процедуры инкапсулируют скрытое сообщение в младшие биты цифрового кода пикселов изображения. Для частотных процедур стеганограмма вставляется в частотную характеристику изображения. Временные процедуры включают следующие методы:

− внедрение цифрового кода сообщения в изображение;

− статистические методы замены: бит изображения заменяется по некоторому статистическому закону;

− частотные процедуры состоят в замене малозначащих частотных характеристик изображения.

IV. Сравнение стеганографических методов

Каждый стеганографичекий метод обладает как сильными, так и слабыми сторонами. Пользователю важно выбрать метод, который в наибольшей степени соответствует поставленной задаче. Все алгоритмы стеганографии должны удовлетворять нескольким основным требованиям. Наиболее важно, чтобы алгоритм давал малозаметное изменение изображения-контейнера. Рассмотрим критерии сравнения:

− *незаметность или уровень восприятия* (нез.). Это главное требование – стеганограмма не должна распознаваться глазом человека. Человек не должен видеть различие между исходным изображением и тем же изображением со вставленным сообщением;

− *вместимость* (вмест.). Это требование определяет размер вставляемого сообщения, который зависит от формата контейнера [2];

− *робастность*. Вставляемое сообщение не должно быть повреждено процессами обработки и передачи, присущими данному формату. Существует два типа робастности:

• *робастность для защиты от статистической атаки* (РПСА). Статистические тесты применяются для выявления наличия стеганограммы в контейнере, это методы статистической обработки данных, которые можно применять как во временной, так и в частотной области;

• *робастность для защиты от целенаправленного повреждения стеганограммы* (РПЦП). Этот вид робастности обусловлен тем, насколько скрытое сообщение зависит от контейнера;

− *способность к обнаружению или скрытность* (СОС). Этот критерий определяет успешность метода скрытия, при распознавании наличия сообщения он обусловливает сложность алгоритма распознавания;

− *вид области* (ВО). Этот параметр указывает на область, в которой применялась стеганография – временная (В) или частотная (Ч);

− *независимость от формата* (НФ). Следует использовать различные форматы файлов. Если партнеры постоянно используют один формат, то это может навести на мысль о тайной переписке [2, 4].

Основные критерий приведены на рис. 1:

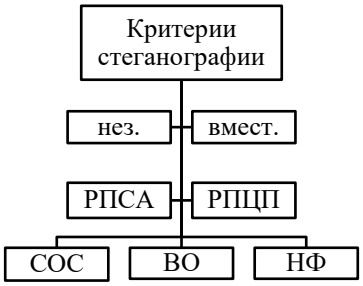


Рисунок 1 – Критерии стеганографии.

В идеальном случае стеганографический алгоритм должен удовлетворять высоким уровням всех критериев. Необходим взвешенный выбор стеганографического метода, который зависит от используемого пользователем приложения. Рассмотрим пригодность различных алгоритмов для форматов файлов:

LSB для BMP. Растровый формат BMP не использует сжатия, поэтому файлы этого формата имеют большой объем. Но для сокрытия сообщения в этих файлах необходим очень большой контейнер.

LSB для JPEG. Распространенный формат JPEG использует 8 битов на каждый цвет RGB. JPEG может скрыть сообщение большого объема.

LSB для цветовой палитры. Формат GIF кодирует пиксел 8 битами, изображение записывается в 256 цветах. Алгоритм LSB скрывает информацию с различными степенями успеха в зависимости от доли изменяемых бит. Необходимо искать равновесие между безопасностью и распознаваемостью

Псевдослучайные перестановки. Метод вставляет биты сообщения с изменением порядка их появления в сообщении, что затрудняет работу по обнаружению и расшифровке сообщения.

Метод с использованием патчей. Недостаток этого метода состоит в том, что в один патч инкапсулируется только один бит [4]. Преимущество этого метода состоит в том, что сообщение распределено по всему изображению, и если один из патчей будет поврежден, то это не принесет больших потерь и сообщение можно восстановить из других патчей.

Метод расширения спектра. Этот метод распределяет сообщение по всему изображению. Такую стеганограмму трудно распознать. Частотная характеристика сообщения обладает гораздо меньшей энергией, чем энергия контейнера. Этот метод имеет большую робастность против атак.

Дискретное косинус-преобразование. Методы области преобразования (частотной области) скрывают сообщение в значительной области изображения, что делает их более робастными по сравнению с методами во временной области, включая сжатие, обрезку и некоторые алгоритмы обработки изображений.

Дискретное вейвлет-преобразование. Инкапсуляция сообщения с помощью ДВП дает хорошие результаты, которые превосходят методы ДКП. Многомасштабный вейвлетный анализ разлагает сигнал в узкие частотные области, что позволяет скрыть сообщение в мелких деталях изображения.

Основные методы стеганографии приведены на рис. 2:

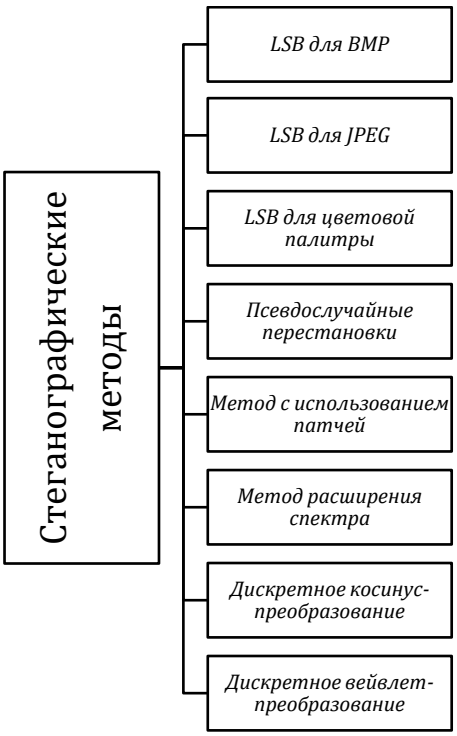


Рисунок 2 – Стеганографические методы.

Выводы

В статье были рассмотрены некоторые из основных методов стеганографии изображений. Все основные форматы графических файлов имеют различные методы сокрытия сообщений со своими сильными и слабыми сторонами. Выбор метода с большой надежностью противостоит методу с высокой скоростью обработки. Например, патч-подход имеет очень высокую устойчивость по отношению к большинству видов атак, но он может скрыть лишь очень небольшое количество информации. Поэтому более разумно скрывать информацию в дополнительных преобразованиях, а не в исходных файлах. Преобразование дискретными вейвлетами более надежно, потому что позволяет скрыть сообщение в области частот. Данная область менее подвержена зрению человека.

Литература

[1] Qiao, Mengyu, Andrew H. Sung, and Qingzhong Liu. "Predicting embedding strength in audio steganography." 9th IEEE International Conference on Cognitive Informatics (ICCI'10). IEEE, 2010.

[2] Ahsan, Kamran, and Deepa Kundur. "Practical data hiding in TCP/IP." Proc. Workshop on Multimedia Security at ACM Multimedia. Vol. 2. No. 7. 2002., p. 783.

[3] Maggi, Federico, Stefano Zanero, and Vincenzo Iozzo. "Seeing the invisible: forensic uses of anomaly detection and machine learning." ACM SIGOPS Operating systems review 42.3 (2008): p. 51-58.

[4] Morkel, Tayana, Jan HP Eloff, and Martin S. Olivier. "An overview of image steganography." ISSA. 2005.

Выбор оптимальных параметров криптосистемы SPHINCS+

Александр Марухненко[1], Геннадий Халимов[1], Антон Янко[2]

1. Кафедра безопасности информационных технологий, Харьковский национальный университет радиоэлектроники, УКРАИНА, г. Харьков, пр. Науки, 14,

2. Кафедра автоматизации и проектирования вычислительной техники, Харьковский национальный университет радиоэлектроники, УКРАИНА, г. Харьков, пр. Науки, 14,
E-mail: oleksandr.marukhnenko@nure.ua

Краткая аннотация – The article describes a structure and parameters of the SPHINCS+ cryptosystem. The right choice of parameters allows you to optimize system performance relative to the time and memory costs. We propose few sets of parameters for 384 and 512-bit security.

Ключевые слова – ЭЦП на основе хеш-функций, SPHINCS, постквантовая криптография, дерево Меркли.

I. Введение

Криптосистемы на основе хеш-функций является перспективным направлением постквантовой криптографии. Особенностью криптосистем этого класса является ограниченное количество подписей, которые могут быть созданы с использованием одного ключа. Первые разработанные алгоритмы позволяли подписывать только одно сообщение, что стало причиной их низкой популярности. Для устранения этого недостатка сначала были предложены хеш-дерева, построенные на основе одноразовых подписей, а затем гипер-деревья, состоящие из хеш-деревьев. В основе алгоритма SPHINCS + лежит именно конструкция гипер-дерева. Использование такой многоуровневой структуры требует установки и обоснование ряда параметров.

Целью работы является проанализировать структуру алгоритма SPHINCS + и предложить определенные параметры для заданных показателей стойкости.

II. Подписи W-OTS и FORS

Общая идея подписи Винтерница (W-OTS – Winternitz One-Time Signature) [1] заключается в следующем: секретным ключом является массив случайных битовых последовательностей, открытым ключом является массив прохешированных *w* (параметр Винтерница) раз элементов приватного ключа, при создании подписи сообщения разбивается на блоки по *w* битов, в конец добавляется контрольная сумма, подпись состоит из элементов приватного ключа, прохешированных определенное количество раз, в зависимости от значения соответствующего блока. Во время проверки элементы подписи «дохешируются» до *w* раз и сравниваются с открытым ключом.

Подпись FORS относится к классу многоразовых подписей с постепенным снижением стойкости. Основная идея этой группы алгоритмов заключается в том, что частный ключ имеет достаточно большой размер, например, 1 мегабайт, при создании подписи раскрывается некая «случайная» часть ключа, таким образом каждая новая подпись увеличивает вероятность подделки. Безопасная количество использований одной ключевой пары определяется требованиями к стойкости системы.

Алгоритм FORS (Forest Of Random Subset) является модификацией алгоритма HORST и используется в схеме SPHINCS + [3]. Суть алгоритма заключается в том, что сообщению однозначно соответствует некоторое подмножество элементов заданного множества (секретного ключа), которое становится подписью, для проверки элементы подписи хешируются и сравниваются с соответствующими элементами из множества хеш-значений (открытого ключа). Особенность FORS заключается в том, что для каждого блока, который подписывается, используют отдельный массив случайных чисел.

III. Дерево и гипер-дерево Меркли

Метод многократного использования пары ключей подпис на основе хеш-функции, предложенный Меркли, базируется на применении так называемых хеш-деревьев или деревьев Меркли [2]. Хеш-деревом называют полное бинарное дерево, в листовые вершины которого помещены хеши от блоков данных, а внутренние вершины содержат хеши от конкатенации значений в дочерних вершинах. Корневой узел дерева содержит хеш от всего набора данных, то есть хеш-дерево является однонаправленной хеш-функцией.

Листьями дерева выступают хеш-значения открытых ключей одноразовых подписей, например ЦП Винтерница. Открытым ключом является корень дерева. Для подтверждения того, что использован в подписи открытый ключ OTS принадлежит данному дереву, к подписи добавляется путь аутентификации - элементы дерева, необходимые для прохождения заданного листа к корню дерева, номер использованного одноразового ключа и сам открытый ключ OTS.

Таким образом проверка подписи состоит из двух этапов - проверка подписи по открытому ключу, аутентификация ключа. В алгоритмах, в которых открытый ключ полностью исчисляется с подписи (WOTS), нет необходимости передавать используемый ключ, позволяющий уменьшить размер подписи.

Описанные ранее механизмы не решают проблемы ограниченного количества использований одной пары ключей. Для FORS стойкость пропорциональна количеству элементов ключа, соответственно большее количество подписей требует увеличения размеров приватного (может быть заменен инициализатор для ГПСЧ) и открытого ключей. Для подписей, основанные на деревьях Меркли, количество

использований также ограничивается при генерации пары ключей, а также для построения дерева на n подписей необходимо сгенерировать n пар ключей одноразовых подписей и провести $2n$ хешированиий внутри дерева, это требует значительных вычислений уже для деревьев с тысячами листьев.

Альтернативным решением является использование структуры гипердерева, представляющая собой дерево из хеш-деревьев, листья деревьев верхнего уровня используются для подписи корней деревьев, лежащих уровнем ниже [4]. Открытым ключом пользователя есть корень верхнего дерева. Такая схема позволяет генерировать дерева низших уровней, не меняя при этом общего ключа. Подпись сообщения включает в себя непосредственно подпись, пути аутентификации на каждом уровне и подписи для промежуточных узлов.

IV. Алгоритм SPHINCS+

Наиболее перспективным ЭЦП на базе хеш-функций в настоящее время является семейство алгоритмов SPHINCS, а именно две независимые модификации первичного подписи SPHINCS - SPHINCS+ [3] и Gravity-SPHINCS, которые были участниками первого тура конкурса постквантовых алгоритмов NIST. Анализ алгоритма SPHINCS+ продолжается в рамках второго тура.

Данный алгоритм использует гипердерево следующего вида: листьями деревьев на всех уровнях являются хеш-значения открытых ключей подписи WOTS+, которые используются для подписи корня соответствующего дерева уровнем ниже, на нижнем уровне ключи используются для подписания открытых ключей алгоритма FORS. Индекс используемой ключевой пары подписи FORS выбирается на основе дайджеста сообщения. Подпись сообщения представляет собой подпись FORS, соответствующая ему подпись WOTS+ из листа гипер-дерева, подписи корней промежуточных деревьев, пути аутентификации и случайная последовательность, которая использовалась во время генерации подписи.

Общесистемные параметры:

- n – параметр безопасности, определяет байтовую длину всех элементов гипер-дерева;
- h – высота гипер-дерева;
- d – количество слоёв гипер-дерева, каждый содержит хеш-деревья высоты $h' = h / d$;
- w – параметр Винтерница;
- k – количество деревьев в FORS;
- t – количество листов одного дерева FORS.

При выборе конкретных параметров алгоритма должен достигаться пространственно-временной компромисс, с одной стороны подпись может быть меньше, но время создания и проверки больше, с другой создания и проверка будут быстрее, но подпись занимает больше памяти. Авторы SPHINCS+ предлагают два набора параметров: «быстрый» и «маленький» для уровней стойкости 128, 192 и 256 бит. Мы предлагаем следующие значения параметров для обеспечения уровней стойкости 384 и 512 бит (Табл.1). Оценка оптимальности предложенных параметров требует дальнейших исследований.

ТАБЛИЦА 1

РЕКОМЕНДУЕМЫЕ ПАРАМЕТРЫ ДЛЯ РАЗНЫХ УРОВНЕЙ СТОЙКОСТИ

	n	h	d	τ	k	w	sec	size
128s	16	64	8	15	10	16	133	8080
128f	16	60	20	9	30	16	128	16976
192s	24	64	8	16	14	16	196	17064
192f	24	66	22	8	33	16	194	35664
256s	32	64	8	14	22	16	255	29792
256f	32	68	17	10	30	16	254	49216
384s	48	64	8	16	23	16	391	59904
384f	48	60	15	10	39	16	390	94800
512s	64	66	6	19	26	16	519	87872
512f	64	65	13	12	42	16	512	148160

Выводы

Цифровые подписи на основе хеш-функций является перспективным направлением пост-квантовой криптографии. Разработчикам криптосистемы SPHINCS+ удалось устранить недостатки, которые имелись в предыдущих алгоритмах. Сложная структура гипер-дерева требует выбора ряда системных параметров, влияющих на стойкость системы, скорость ее работы и размеры подписи, поэтому сложно однозначно выбрать лучший набор составляющих, поскольку в зависимости от конкретного применения требования могут меняться. Мы предложили значения параметров, которые могут быть применены в криптосистеме SPHINCS+ с устойчивостью 384 и 512 бит.

Литература

[1] Andreas Hülsing. W-OTS+ – shorter signatures for hash-based signature schemes. In Amr Youssef, Abderrahmane Nitaj, and Aboul-Ella Hassanien, editors, Progress in Cryptology – AFRICACRYPT 2013, volume 7918 of LNCS, pages 173–188. Springer, 2013.

[2] Ralph Merkle. A certified digital signature. In Gilles Brassard, editor, Advances inCryptology – CRYPTO ’89, volume 435 of LNCS, pages 218–238. Springer, 1990.

[3] Daniel J. Bernstein, Christoph Dobraunig, Maria Eichlseder and others. SPHINCS+ – Submission to the NIST’s post-quantum cryptography standardization process, 2017.

[4] Andreas Hülsing, Lea Rausch, and Johannes Buchmann. Optimal parameters for XMSSMT. In Alfredo Cuzzocrea, Christian Kittl, Dimitris E. Simos, Edgar Weippl, and Lida Xu, editors, Security Engineering and Intelligence Informatics, volume 8128 of Lecture Notes in Computer Science, pages 194–208. Springer Berlin Heidelberg, 2013.

Системний аспект дослідження системи управління інформаційною безпекою

Володимир Мохор[1], Василь Цуркан[2]

1. Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова Національної академії наук України, УКРАЇНА, м. Київ, вул. Генерала Наумова, 15, E-mail: v.mokhor@gmail.com

2. Кафедра кібербезпеки і застосування інформаційних систем і технологій, Інститут спеціального зв'язку та захисту інформації Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського», УКРАЇНА, м. Київ, вул. Верхньоключова, 4, E-mail: v.v.tsurkan@gmail.com

Коротка анотація – The system of information security management is considered as a proactive factor in providing cybersecurity. Typical examples of aspects of its research are analyzed. In particular, the formation of a system of concepts in the field of information security management, the establishment of design characteristics of the construction and evaluation of the maturity of the system, the assessment of information security risk were analyzed. Most of them are oriented towards the separate side of the information security management system. To overcome this problem, its research in the systemic aspect is proposed. This made it possible to view the information security management system as a single entity, which is displayed by many elements. Separate structural and functional links for each of them. It is focused on further perspectives as a description of the theoretical foundations of information security management system systematic studies.

Ключові слова – кібербезпека, інформаційна безпека, управління інформаційною безпекою, система управління інформаційною безпекою, системні дослідження, системний підхід, теорія систем, системний аналіз.

I. Вступ

Однією з передумов забезпечення кібербезпеки є розробляння в організації системи управління інформаційною безпекою [1]. Завдяки цьому гарантується неперервне ідентифікування, аналізування, атестування і обробляння ризиків. Ці ризики обумовлені як діяльністю організації, так і наданням нею послуг у кіберпросторі [1, 2]. Такий підхід сприяє розширянню сфери її відповідальності та дозволяє унеможливити збільшення ризиків кібербезпеки. Тому система управління інформаційною безпекою розглядається як один з найбільш дієвих проактивних факторів забезпечення кібербезпеки з прийнятним рівнем ризику діяльності організації у кіберпросторі [2].

Вимоги до розробляння, впроваджування, функціонування, контролювання, переглядання і вдосконалювання системи управління інформаційною безпекою викладено в міжнародних стандартах серії ISO/IEC 27k. Зокрема, основним з них є гармонізований в Україні ДСТУ ISO/IEC 27001 [1], який на практиці доповнюється настановами міжнародних стандартів серії ISO 31k.

II. Аспекти досліджень системи управління інформаційною безпекою

Система управління інформаційною безпекою є частиною загальної системи управління організацією у контексті оцінювання ризиків [2]. При цьому характерною особливістю означених систем є багатоаспектність їх досліджень. Розглянемо приклади окремих аспектів [3-9]:

1. Аналізування міжнародних стандартів серії ISO/IEC 27k [3]. Зокрема, історії розвитку означених нормативно-провавих документів, особливості використання, перспективи розвивання і, як наслідок, гармонізування в Україні. При цьому акцентується увага на особливостях адаптування і узгодження з вітчизняною нормативно-правовою базою.

2. Формування системи понять у галузі управління інформаційною безпекою для забезпечування усталеного використання термінів і визначень [4]. Виокремлюються проблеми їх використання на практиці через неузгодженість і неоднозначність їх трактування, відсутність взаємозв'язку між ними.

3. Встановлення проектних характеристик систем управління інформаційною безпекою для підвищення ефективності процесу проектування означених систем [5]. Для цього доведено аналогію системи управління інформаційною безпекою з системою масового обслуговування. Як наслідок, показано недостатність застосування прийнятного ризику як проектної характеристики.

4. Визначення прийнятного рівня ризику інформаційної безпеки для його атестування [6]. Зокрема, поділу оцінок ризику інформаційної безпеки на прийнятні та неприйнятні. Як наслідок обґрунтування необхідності прийняття рішення про обробляння ризику. Аналізування його щодо застосовності як проектної характеристики побудови системи управління інформаційною безпекою [5, 6].

5. Оцінювання ризику інформаційної безпеки для прийняття рішення про необхідність і/як наслідок обирання заходів і засобів його обробляння [7]. Для цього використовуються методи визначення як кількісних, так і якісних оцінок. При цьому складність їх використання на практиці обумовлена, з одного боку, відсутністю статистичних даних про реалізації загроз. З другого боку, складністю зіставлення і відтворювання оцінок ризику інформаційної безпеки.

6. Оцінювання впливу людського фактору на забезпечення конфіденційності, цілісності та доступності інформації [8]. Вплив означеного фактору розглядається як з середини (інсайдерська загроза), так і ззовні організації (загроза використання соціальної інженерії).

7. Оцінювання зрілості системи управління інформаційної безпеки для пошуку шляхів підвищення її рівня до бажаного [9]. Для цього виокремлюються відповідні рівні (наприклад, від 0 до 4) та встановлюється, наприклад, наявність/відсутність відповідних правил, процедур на кожному з них.

8. Моделювання системи управління інформаційною безпекою за організаційно-технічним підходом [10]. При цьому означена система здебільшого розглядається у аспекті підготовлення документів на кожному з етапів її розроблення. Водночас залишаються не вирішеними питання стосовно елементів, взаємозв'язків між елементами системи управління інформаційною безпекою.

III. Системний аспект дослідження системи управління інформаційною безпекою

За системним аспектом система управління інформаційною безпекою розглядається як цілісний об'єкт [11]. Це дозволяє з одного боку врахувати багатоаспектність її дослідження. Тоді як з іншого виокремити елементи системи управління інформаційною безпекою та, як наслідок, встановити між ними структурні та функціональні зв'язки.

У цьому випадку єдність складу, організації, структури системи управління інформаційною безпекою характеризується внутрішньою будовою. Складом описується сукупність усіх елементів і визначається багатоманітність і складність системи. До того ж від складу залежить її природа. Внаслідок цього можлива зміна властивостей системи залежно від зміни її складу. Такі зміни враховуються завдяки таким характеристикам як структура та організація.

Цілісність об'єкта забезпечується сукупністю стійких зв'язків між елементами, що утворює структуру системи управління інформаційною безпекою. Виокремлення структури дозволяє визначати властивості системи через її будову. Завдяки цьому пояснюється діяльність зі забезпечення інформаційної безпеки.

Внутрішня упорядкованість елементів, сукупність процесів, що визначаються взаємозв'язки між елементами, системи управління інформаційною безпекою утворюють її організації. Організація системи відображає як статистичний, так і динамічний контексти. Вона описується через структуру, орієнтації і розподіл функцій.

Висновок

Використання міжнародних стандартів серій за ISO/IEC 27k і ISO 31k дозволяє проактивно забезпечувати кібербезпеку. При цьому за основу береться організаційно-технічний підхід і, як наслідок, дослідження системи управління інформаційною безпекою характеризуються багатоаспектністю. Це призводить до проблеми виокремлення елементів означеної системи і встановлення між ними структурних і функціональних зв'язків. Для подолання цієї проблеми запропоновано досліджувати систему управління інформаційною безпекою в системному аспекті.

У подальших перспективах буде розглянуто теоретичні основи системних досліджень системи управління інформаційною безпекою.

Література

[1] ISO/IEC 27032:2012. Information technology. Security techniques. Guidelines for cybersecurity. – [First edition 2012-07-15]. – Geneva, 2012. – 50 p.

[2] ДСТУ ISO/IEC 27001:2015. Інформаційні технології. Методи захисту системи управління інформаційною безпекою. Вимоги (ISO/IEC 27001:2013; Cor 1:2014, IDT). – [Чинний від 2015-12-18]. – Київ, 2016. – 22 с.

[3] Humphreys E. Information Security Management System Standards / E. Humphreys // Datenschutz und Datensicherheit. – January 2011. – Vol. 35, Iss. 1. – pp. 7-11. – doi: 10.1007/s11623-011-0004-3.

[4] Мохор В.В. Формування системи понять у галузі управління інформаційною безпекою / В.В. Мохор, О.М. Богданов, О.О. Бакалинський, В.В. Цуркан // Інформаційні технології та безпека : засади забезпечення інформаційної безпеки : матеріали XIV міжнародної науково-практичної конф. Київ : ІПРІ НАН України, 2014. С. 98-99.

[5] Мохор В.В. Представлення оцінок ризиків інформаційної безпеки картою ризиків / В.В. Мохор, О.О. Бакалинський, В.В. Цуркан // Information technology and security. – 2018. – Vol. 6, Iss. 2. – pp. 94-104. – doi: 10.20535/2411-1031.2018.6.2.153494.

[6] Безштанько В.М. Диофантов метод определения частости нанесения ущерба вследствие реализации угрозы информационной безопасности / В.М. Безштанько, В.В. Цуркан // Захист інформації. – 2013. – Т. 15, № 4. – С. 278-283. – doi: 10.18372/2410-7840.15.5707.

[7] Корченко О.Г. Метод оцінювання ризиків інформаційної безпеки на основі відкритих баз даних уразливостей / О.Г. Корченко, С.В. Казмірчук // Захист інформації. – 2016. – Том 22, № 2. – С. 214-224. – doi: 10.18372/2225-5036.22.10716.

[8] Reza A. A Conceptual Framework to Analyze Human Factors of Information Security Management System (ISMS) in Organizations / A. Reza, S. Islam, H. Mouratidis // International Conference on Human Aspects of Information Security, Privacy, and Trust (HAS 2014). – Heraklion, Crete, Greece, 22-27 June, 2014. – pp. 297-305. – doi: 10.1007/978-3-319-07620-1_26.

[9] Proença D. Information Security Management Systems – A Maturity Model Based on ISO/IEC 27001 / D. Proença, J. Borbinha // International Conference on Business Information Systems (BIS 2018). – Berlin, Germany, 18-20 July, 2018. – pp. 102-114. – doi: 10.1007/978-3-319-93931-5_8.

[10] Домарев Д.В. Метод інформаційно-аналітичної підтримки управління інформаційною безпекою на основі системного підходу / Д.В. Домерев // Захист інформації. – 2014. – Т. 16, № 2. – С. 146-158. – doi: 10.18372/2410-7840.16.6508.

[11] Згуровский М.З. Системный анализ : проблемы, методология, применение / М.З. Згуровский, Н.Д. Панкратова. – К. : Наукова думка, 2005. – 743 с.

Дослідження можливості проведення invariant subspace атаки на енергоефективний алгоритм Midori

Олексій Наумов[1], Віктор Руженцев[2]

1. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, м. Харків, пр. Науки, 14, E-mail: oleksii.naumov@nure.ua

2. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, м. Харків, пр. Науки, 14, E-mail: viktor.ruzhentsev@nure.ua

In the past few years, lightweight cryptography has become a popular research discipline with a number of ciphers and hash functions proposed. The designers' focus has been predominantly to minimize the hardware area, while other goals such as low latency have been addressed rather recently only. However, the optimization goal of low energy for block cipher design has not been explicitly addressed so far. At the same time, it is a crucial measure of goodness for an algorithm. Indeed, a cipher optimized with respect to energy has wide applications, especially in constrained environments running on a tight power budget such as Internet of Things.

Midori, lightweight блоковий шифр, AES, слабкі ключі, бітові перестановки, S-box.

I. Вступ

В сучасному світі з кожним роком все більше набуває популярності Інтернет Речей (Internet of Things, IoT). З'являється велика кількість розумних пристроїв вдома, в лікарні, у школі. Виходячи з цього виникає необхідність захисту цих пристроїв від зловмисників. Але через обмеженість в продуктивності, пам'яті чи енергоспоживанні не представляється можливим використання звичайних криптографічних засобів. Рішенням цієї проблеми є легковагова (lightweight) криптографія.

Легковагова криптографія – це розділ криптографії, який спрямований на розробку алгоритмів для використання в пристроях, які не здатні забезпечити виконання більшості існуючих алгоритмів або мають обмежені ресурси (пам'ять, потужність, енергоспоживання). Одним з найбільш енергоефективних та популярних алгоритмів є блочний симетричний шифр Midori.

II. Короткий опис Midori

Midori – це сімейство двох блокових шифрів: Midori64 і Midori128. Вперше було представлено світові на міжнародній конференції ASIACRYPT у 2015 році. Сімейство шифрів відноситься до SPN алгоритмів та має AES – подібну структуру. Обидва шифри приймають 128-бітові ключі і мають різний розмір блоку, 64 біти для Midori64 і 128 біт для Midori128. На виході отримуємо шифротекст тієї ж довжини, що і вхідний блок.

Циклічна функія Midori складається з наступних чотирьох операцій:

1) SubCell - процедура підстановки. Використовується два типи 4-бітних S-боксів, Sb_0 і Sb_1 $\{0, 1\}^4 \rightarrow \{0,1\}^4$ (див. Табл. 1). Sb_0 використовується в Midori64, а Sb_1 – в Midori128.

ТАБЛИЦЯ 1

Таблиця підстановок

S	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	C	A	D	3	E	B	F	7	8	9	1	5	0	2	4	6
2	1	0	5	3	E	2	F	7	D	A	9	B	C	8	4	6

Midori128 використовує чотири різних 8-бітових S-бокси SSb_0, SSb_1, SSb_2 і SSb_3, де SSb_0, SSb_1, SSb2, SSb_3: $\{0, 1\}^8 \rightarrow \{0,1\}^8$. Математично, кожен SSb_i складається з вхідних і вихідних бітових перестановок і двох Sb_1, як показано на Рис. 1.

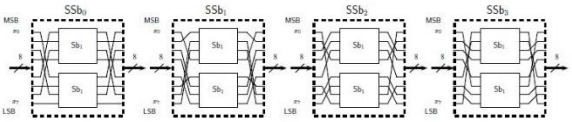


Рисунок 1 – Схема підстановки в Midori128

2) ShuffleCell - процедура циклічного зсуву. Це перетворення виконує розподілення бітів кожного рядка матриці стану серед всієї матриці. Кожна клітинка стану переставляється так:

$$(s_0, s_1, \ldots, s_{15}) \leftarrow (s_0, s_{10}, s_5, s_{15}, s_{14}, s_4, s_{11}, s_1, s_9, s_3, s_{12}, s_6, s_7, s_{13}, s_2, s_8) \quad (1)$$

3) MixColumn виконує множення кожного стовпця стану на матрицю M.

$$M = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \quad (2)$$

Операції множення між матрицею і вектором виконуються над полем $GF(2^m)$.

4) KeyAdd - процедура додавання ключа. Для Midori64 128-бітний секретний ключ K розділяється на два 64-бітні ключі K_0 і K_1. Тоді $WK = K_0 \oplus K_1$.

$$RK_i = K_{(i \bmod 2)} \oplus \alpha_i \quad (3),$$

де $0 \le i \le 14$. Для Midori128, WK = K, $RK_i = K \oplus \beta_i$, де $0 \le i \le 18$.

α_i та β_i - раундові константи, наведені у роботі [1].

Перед першим раундом застосовується додаткова операція KeyAdd, а в останньому раунді операції ShuffleCell та MixColumn відсутні. Загальна кількість циклів для Midori64 та Midori128 – 16 та 20 відповідно.

Загальна процедура зашифрування виглядає наступним чином. Блок вхідного тексту складається з таємним ключем. Потім R - 1 разів, де R – кількість циклів, відбуваються наступні дії: підстановка (SubCell), зсув рядків (ShuffleCell), перемішування стовпців (MixColumn) та додавання циклового ключа. На останньому циклі відбувається лише підстановка та додавання таємного ключа. Схема зашифрування для Midori64 представлена на Рис. 2.

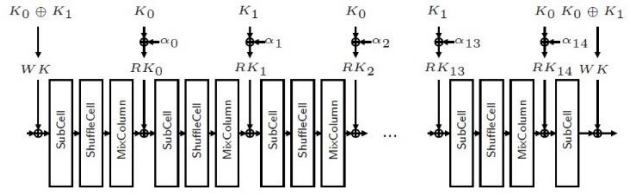


Рисунок 2 - Схема зашифрування Midori64

III. Недоліки Midori

Основним завданням під час розробки будь-якого легковагового блокового симетричного шифру є створення енергоефективного, стійкого та невибагливого до ресурсів алгоритму. Досягти одразу всі три цілі одночасно є досить складним завданням. Розробники сімейства шифрів Midori спробували це зробити. Midori вважається найбільш енергоефективним шифром серед конкурентів. Але окрім цього знайшовся ряд недоліків. До основних з яких відноситься наявність класу слабких ключів розміром 2^{32} [2]. Враховуючи операції, які використовуються в цикловій функції та вищеперераховані особливості можна підібрати відкритий текст, що стає підґрунтям для подальшого проведення атаки invariant subspace. Слід зазначити, що подібна властивість притаманна лише версії шифру з довжиною блоку 64 біти. Її детальний опис наведений нижче.

Нехай ключ складається з напівбайт, які відносяться до множини, що складається з двох елементів 0 та 1. Відкритий текст складається з множини напівбайт елементів 8 та 9. Оскільки ключ належить до множини значень {0, 1}, то і всі циклові ключі будуть відноситись до тієї ж множини значень, дивись (3).

Будь-які операції складання з такими ключами будуть змінювати блок стану, що належить до множини значень {8, 9}, але він і далі буде належати ті й множині, оскільки складання з ключем буде змінювати лише молодший біт кожного напівбайта.

Операція SubCell не спричинить зміни стану, оскільки згідно з Табл.1 після підстановки $8 \rightarrow 8$, а $9 \rightarrow 9$.

Операція ShuffleCell лише перемішує напівбайти між собою, а значить блок стану все ще буде належати множині {8, 9}.

Під час MixColumn виконується перемноження кожного стовпця матриці стану на матрицю (2). Оскільки вона складається лише з 0 та 1, буде справедливим твердження, що блок тексту все ще буде належати множині {8, 9}.

Отже, використання будь-якого слабкого ключа веде до того, що шифротекст буде належати ті й же множині значень, що й відкритий текст. Кількість циклів або порядок операцій не порушують цю властивість [3].

IV. Можливі протидії проведення invariant subspace атаки

Оскільки проведення invariant subspace атаки з відновлення ключа є можливим незалежно від кількості циклів, то лише збільшити їх кількість недостатньо. Отже необхідно змінити саму циклову функцію або механізм генерації циклових ключів. Нижче наведені декілька основних напрямків оптимізації алгоритму.

Перший напрямок це заміна самого S-боксу. Нова таблиця підстановок повинна бути уважно підібрана, без можливості переходу значення напівбайт в себе ж.

Другий напрямок це заміна циклових констат, які беруть участь у генерації циклових ключів. Наприклад, можна змінити один напівбайт в кожній константі на номер циклу.

Останній напрямок це застосування бітових перестановок під час операції SubCell. В Midori вже реалізований подібний механізм для генерації таблиць підстановок для 128-бітової версії шифру (Рис. 1).

Висновки

В цій роботі був досліджений блоковий симетричний шифр Midori. Окрім принципів його побудови був описаний важливий недолік цього шифру, що може слугувати підґрунтям для проведення invariant subspace атаки із відновлення ключа. Були запропоновані шляхи підвищення стійкості алгоритму. Цілями подальших робіт буде практичне застосування цієї атаки, впровадження запропонованих шляхів рішення існуючих проблем та аналіз їх впливу на роботу алгоритму(швидкодію, стійкість).

Перелік посилань

[1] Subhadeep Banik. Midori: A Block Cipher for Low Energy (Extended Version)/ Andrey Bogdanov, Takanori Isobe, Kyoji Shibutani, Harunaga Hiwatari, Toru Akishita and Francesco Regazzoni – ASIACRYPT-2015, pages 10-14.

[2] Xiaoyang Dong Cryptanalysis of Reduced-Round Midori64 Block Cipher / Xiaoyang Dong, Yanzhao Shen - ISCISC 2017, pages 2-4.

[3] Jian Guo Invariant Subspace Attack Against Full Midori64 / Jian Guo, Jeremy Jean, Ivica Nikolic, Kexin Qiao, Yu Sasaki, Siang Meng Sim - EUROCRYPT 2017, pages 5-7.

Аналіз захищеності системи Android для використання в корпоративному сегменті

Костянтин Нечволод[1], Олександр Сєвєрінов[2]

1. Кафедра безопасности информационных технологий, Харьковский национальный университет радиоэлектроники, УКРАИНА, г. Харьков, пр. Науки, 14, E-mail: kostiantyn.nechvolod@nure.

2. Кафедра безопасности информационных технологий, Харьковский национальный университет радиоэлектроники, УКРАИНА, г. Харьков, пр. Науки, 14, E-mail: oleksandr.sievierinov@nure.ua

Краткая аннотация – Android adoption has increased rapidly over the last few years, becoming the go-to OS for many organisations the world over. Due to the diversity of the platform and flexibility of form factor, application and budget, Android is making a huge impact on how employees undertake their daily responsibilities.

Ключевые слова –android, google play protect, google, mdm, безпека.

I. Вступ

Android – це надійна операційна система, яка встановлена на дуже велику кількість пристроїв, починаючи від мобільних телефонів і закінчуючи головними пристроями в автомобілі. Вона базується на таких основах, як відокремлення системних процесів, архітектурі довіреної ОС, а також вона використовує потужний аналізатор загроз який використовує машинне навчання та хмарні обчислення для визначення загроз за допомогою великої бази знань компанії Google.

Враховуючи ці фактори, підприємствам слід звернути увагу на операційну систему Android при впроваджені концепцій MDM та BYOD.

II. Вирішення проблеми та результати

Багато підприємств скоро розширять кількість мобільних пристроїв, які будуть використовуватися в корпоративній сфері за допомогою сучасних підходів та концепцій як MDM або BYOD. [1]

Але, незважаючи на те, що мобільність підприємств зростає, безпека даних залишається головною перешкодою на шляху впровадження нових технологій використання мобільних пристроїв.

Зараз одними з найпоширеніших загроз для мобільних пристроїв є:

– використання для доступу до глобальної мережі через незахищені точки доступу Wi-Fi, в яких можлива атака MITM(людина посередині);

– можливість крадіжки або втрати пристрою, яка дозволить зловмиснику отримати повний фізичний доступ до пристрою;

– шкідливе програмне забезпечення, яке може бути встановлено не навмисно самим користувачем і яке буде тихо збирати усю потрібну зловмиснику інформацію та навіть робити фото, відео та аудіо фіксацію усього, що трапляється навколо пристрою.[2]

Однак остання загроза є найменш вірогідною через сам принцип побудови системи Android, де кожен додаток виконується відокремлено від інших та отримати доступ до даних іншої програми без отримання прав root неможливо.

Сама думка про те, що доступ до конфіденційної інформації може бути не в корпоративній мережі викликає дуже багато сумнівів щодо доцільності використання подібних систем у фахівців з IT-безпеки. Тому перш за все, перед початком впровадження системи необхідно зважити всі переваги та ризики для бізнесу і вже потім приймати подальші рішення.[4]

Принести власний пристрій (BYOD) є поширеною концепцією у більшості підприємств, наприклад 84% опитаних компанією IDC в США заявили, що вони дозволяють хоча б певний ступінь використання персональних пристроїв на робочому місці. Використання персональних пристроїв на робочому місці - це тенденція, яка торкається більшості IT-підрозділів[3]. Хоча BYOD допомагає підприємствам скоротити витрати і може покращити задоволеність працівників, це також може бути проблемою безпеки.

За даними IDC, у підприємств з великим процентом використання BYOD виникли частіші проблеми з безпекою. Серед організацій с переважно більшим розгортанням BYOD, інциденти з мобільною безпекою були на 10-12% частішими. Серед фірм, що обмежують або забороняють BYOD, рівень відповідей на питання безпеки був нижчим за середній на 7%. Розмита межа робочих / особистих технології поширюється за межі пристроїв на додатки та хмарні послуги. Користувачі зазвичай залучають до роботи особисті або бажані мобільні додатки, хмарне сховище, програми SaaS та інші технології, які, на їх думку, роблять їх більш продуктивними. Відповідно до опитування менеджерів IT Mobile Mobile IDC за 2017 рік (спонсор Google) більше 70% підприємств США та Європи допускають певну тіньову IT у своїх організаціях. Ці фірми розуміють, що все дозволений, але пильний підхід до некорпоративних хмарних та програмних технологій, які використовують працівники, може сприяти підвищенню продуктивності та підвищити задоволеність користувачів. Незважаючи на те, що пристрої під керуванням операційної системи Android мають потужні основи безпеки, цілий ряд хмарних сервісів підтримує пристрої Android для подальшого покращення та забезпечення загальної безпеки платформи. Google Mobile Services (GMS) - це пакет програм, що ліцензуються Google сторонніми виробниками програмного забезпечення та партнерів Android, що дозволяє легко контролювати попередню інсталяцію таких програм, як Gmail, Карти, YouTube, Google Play Store та інші [2]. Менш очевидним для користувачів Android є основні можливості безпеки, які поставляються із GMS. Будь-який пристрій, що має ліцензію на GMS, також має функції сканування програмного забезпечення на основі пристроє та хмарно заснованих засобів для безпеки, а саме базовий набір послуг та функцій, який називається Google Play Protect. Вони варіюються від сканування на пристрої для використання потенційно небезпечних додатків (PHA) та експлуатування додатків, до програми Find My Device (раніше «Диспетчер пристроїв Android»)

для пошуку втрачених чи вкрадених пристроїв та виявлення наявності прав рут.

Google Play Protect включає набір API, які взаємодіють та обробляють інформацію між додатками на пристроях за допомогою вбудованих функцій захисту пристроїв та хмарні служби безпеки. У 2017 році Google Play Protect автоматично відключив PHA з приблизно 1 мільйона пристроїв. Google рецензує всі додатки, перш ніж публікувати їх у магазині Google Play. Окрім перегляду програм, поданих у Google Play, його хмарні системи шукають додатки у загальнодоступних джерелах. Google Play Protect також розглядає програми, які знаходить поза Google Play для PHA. Google Play Protect охоплює всі засоби безпеки, які роками захищають безпеку пристроїв користувачів Android. Наприклад, служба Verify Apps у Google Play Protect сканує програми для PHA, перш ніж користувачі встановлять їх, незалежно від їх походження.[2] Послуга Verify Apps виконує періодичну перевірку на всьому пристрої, яка інспектує додатки перед встановленням та виконує регулярні сканування всіх встановлених програм. Якщо PHA знайдено, сповіщення просить користувача видалити його. У випадках, коли PHA не надає можливих переваг користувачам, Google Play Protect може видалити PHA з уражених пристроїв та заблокувати майбутні встановлення.

AutoScan - це ще одна послуга, яка щодня перевіряє пристрої Android на предмет наявності PHA та інших ознак підробки. (Цей сервіс минулого року сканував майже 800 мільйонів пристроїв на день таких, як смартфони, планшети та телевізори з ОС Android.) AutoScan працює разом з Verify Apps як частина багатошарового підходу сканування Google до програмного забезпечення та безпеки Android. Якщо AutoScan виявить PHA або інші індикатори ризику на пристрої, це може викликати додаткове локальне сканування програм для подальшого вивчення проблеми. Хоча архітектура безпеки пристрою / хмари, що базується на хмарі, забезпечує надійний захист, вона значною мірою працює поза зони видимості для кінцевих користувачів. Google Play Protect був представлений у травні 2017 року, щоб розкрити цю функціональність та допомогти користувачам зрозуміти «стан здоров'я» своїх пристроїв. Google Play Protect забезпечує активні сповіщення користувачів Android про те, коли відбувається сканування на пристрої, стан безпеки кожного додатка, який переглядають або завантажують із магазину Google Play, та загальний стан додатків на пристрою. Швидке поширення оновлень програмного забезпечення є проблемою в такій різноманітній екосистемі, як Android. Google розповсюджує критичні оновлення безпеки через GMS, коли ця служба працює незалежно від оператора або версії програмного забезпечення Android.[3] Це може відправити нове програмне забезпечення, як тільки виявляться нові загрози. Крім критичних оновлень, часті оновлення ОС також забезпечують безпеку та продуктивність користувачів. З цією метою Google запустив Project Treble у 2017 році, щоб зробити його основні служби ОС більш модульними та однорідними, що дозволяє легше впроваджувати оновлення програмного забезпечення для операторів та виробників пристроїв. Treble змінює фреймворк для взаємодії компонентів операційної системи Android та компонентів постачальника пристроїв (чіпи від Qualcomm, MediaTek тощо).[3]

Це вирішує проблему швидких оновлень шляхом впорядкування процесу, коли виробники девайсів працюють з виробниками чіпів для перевірки нових оновлень ОС Android. Treble створює стандарт інтерфейсу постачальника для компонентів нижчого рівня для взаємовії з ОС. Виробники пристроїв та виробники чіпів більше не повинні переробляти низькорівневий код при кожному випуску, прискорюючи час оновлення.

Це посилює безпеку, оскільки знімається потреба у пряму доступу до драйверів ядра, які керують відтворенням медіа, що забезпечує більш надійну «пісочницю» та ускладнює скомпроментуваня фреймворку для використання ядра. (Однак підприємства можуть також навмисно затримувати оновлення ОС на пристроях, які рекомендовано програмою Android Enterprise, щоб дозволити IT-командам тестувати та оновлювати додатки для роботи на останній версії Android.).

Висновки

Операційна система Android вже давно лишилась старих проблем з безпекою. На сьогоднішній день безпеці цієї ОС приділяють багато часу та уваги, що безумовно грає велику роль в використанні пристроїв на базі Android не лише в приватній сфері, а ще і в корпоративному сегменті. Окрім вбудованих засобів безпеки, що працюють на рівні ядра системи, використання хмарних сервісів для постійного сканування системи на наявність потенційно небезпечного ПЗ, виявлення загроз ще на етапі завантаження додатків до магазину Google Play, наявність сервісу для пошуку втраченого пристрою та постійний випуск патчей безпеки для ОС дозволяють зробити висновки, що базовий варіант системи Android є досить безпечним для використання навіть в корпоративному сегменті.

Література

[1] Нечволод К.В. Аналіз безпеки даних в EMM системах / К.В. Нечволод, О.В. Сєвєрінов, А.В. Власов // Системи управління, навігації та зв'язку. – Полтава: ПНТУ. - 2019. – Вип. 3(55). – С. 131-134

[2] Android Enterprise [Электронный ресурс]. – режим доступа к журн.: https://www.android.com/enterprise/

[3] .IDS. Mobile Security at Enterprise Scale, 2019.

[4] Нечволод К.В. Аналіз безпеки даних на основі платформи Samsung Knox / К.В. Нечволод, О.В. Сєвєрінов // Комп'ютерні та інформаційні системи і технології. Третя міжнародна науково-технічна конференція. Збірник наукових праць. Х: ХНУРЕ, 2019.– С. 80-81.

Ризики інформаційної безпеки в хмарних сервісах

Денис Нікішин, Олександр Федюшин

Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА,
м. Харків, пр. Науки, 14,
E-mail: mus1kdamage@gmail.com,
oleksandr.fediushyn@nure.ua

Коротка анотація – Models and methods of expert audit of information security in the cloud computing system are described. The purpose of the work is to increase the speed of response to incidents of information security breaches in the cloud services provision system by conducting an operational audit.

Ключові слова – Аудит інформаційної безпеки, хмарні послуги, хмарні технології, архітектура хмарних обчислень.

I. Вступ

На сьогоднішній день в індустрії інформаційних сервісів спостерігається стрімкий розвиток інформаційних систем, побудованих на основі хмарних технологій (ІСХТ).

У зв'язку з тим, що, найчастіше, компанії-споживачі хмарних послуг обробляють в хмарних середовищах критичні для них інформаційні активи, питання забезпечення інформаційної безпеки ІСХТ стає найбільш актуальним для споживача хмарних послуг.

Хоча постачальники, обслуговуючи підприємства з оборотом в мільярди доларів, роблять все можливе для забезпечення максимальної безпеки середини хмари, особливості побудови хмарної інфраструктури, використання технології віртуалізації і пов'язана з нею можливість порушення ізоляції віртуальних машин, а також паралельна обробка великих обсягів даних різних споживачів хмарних послуг, дозволяють говорити про нові потенційно можливі загрози інформаційної безпеки (ІБ), які будуть специфічними для хмари, обчислень.

II. Актуальність проблеми

Об'єктивна оцінка ступеня захищеності даних споживача хмарних послуг в хмарній інфраструктурі постачальника здійснюється в ході проведення аудиту інформаційної безпеки. В результаті аудиту інформаційної безпеки ІСХТ можна виявити, наскільки раціонально вирішені питання безпеки інформації та контролю доступу в хмарному середовищі, визначити, як мінімізувати ризики при обробці в ІСХТ конфіденційної інформації споживача хмарних послуг, локалізувати слабкі місця в системі забезпечення інформаційної безпеки і сформулювати рекомендації щодо шляхів вирішення існуючих проблем безпеки в системі. Крім того, споживач зацікавлений у визначенні постачальником конкретного об'єктивного механізму для оцінки якості пропонованої хмарної послуги.

Разом з тим, для забезпечення збереження конфіденційної інформації, що обробляється в хмарному середовищі, існуюча система забезпечення інформаційної безпеки ІСХТ повинна періодично піддаватися незалежному аудиту, який відповідно до вимог міжнародних стандартів, є одним з обов'язкових етапів життєвого циклу будь-якої інформаційної системи.

III. Аналіз проблеми

Багато експертів застерігають від надмірної довіри до заяв постачальників про безпеку хмарних обчислень [1], так як широко поширена думка, що споживач хмарних послуг має той рівень захищеності у хмарному середовищі, який забезпечується постачальником.

У США асоціація Cloud Security Alliance випустила Cloud Controls Matrix. Цей документ являє собою перелік існуючих технологій інформаційної безпеки, які можуть бути використані в хмарних сервісах. Хоча деякі фахівці вважають, що для управління ІБ при побудові хмари SaaS можуть бути використані стандарти ISO 27001 та ISO 27002, все ж необхідна розробка спеціальних стандартів для хмарних обчислень.

Для хмарних середовищ загрози віддаленого злому і зараження шкідливим кодом дуже значимі із-за паралельного існування безлічі віртуальних машин. Відмінною особливістю віртуальної машини, яку потрібно враховувати, є можливість її зараження у вимкненому стані [1], якщо є доступ до сховища образів віртуальних машин через мережу Інтернет. Тому особлива увага повинна бути приділена приватним політикам безпеки, особливо політикам розмежування доступу, інформаційних систем, побудованих на основі технологій хмарних обчислень.

Також варто відзначити, що спеціалізовані засоби забезпечення інформаційної безпеки, які застосовуються у сучасних інформаційних системах, знижують ефективність і швидкість обробки інформації клієнта у хмарі, а хмарні продукти безпеки, що допомагають подолати цю проблему, в даний час в нашій країні не сертифіковані.

Інформаційна безпека повинна забезпечуватися на всьому ланцюжку, включаючи постачальника хмарного рішення, споживача та пов'язуючі їх комунікації [2]. Споживач хмарних послуг зобов'язаний вводити до своєї системи відповідну політику безпеки, яка виключає передачу прав доступу до інформації, наданої постачальником, третім особам. Хмари не скасовують необхідність розробки і впровадження політики безпеки у сегменті споживача і використання сервісів безпеки, покликаних гарантувати захист призначених для користувача робочих місць на стороні споживача хмарних послуг.

IV. Розв’язання проблеми

При побудові системи забезпечення інформаційної безпеки системи хмарних обчислень (СХмО) необхідно брати до уваги загрози порушення інформаційної безпеки конкретного об'єкта захисту. В даній роботі складено перелік загроз в системі хмарних обчислень (СХмО) на основі моделі надання послуг SaaS.

1. Загрози безпеки інформації для споживача хмарних послуг:

– загрози безпеки інформації, пов'язані з невизначеністю відповідальності. Дані загрози з'являються у зв'язку з відсутністю чіткого поділу відповідальності в частині забезпечення інформаційної безпеки між споживачем і постачальником послуг SaaS, що є причиною невиконання деяких заходів щодо захисту інформації;

– загрози безпеці інформації, пов'язані з втратою управління. Використання хмарних послуг має на увазі передачу споживачем хмарних послуг частини функцій управління своєю інформаційною системою постачальнику хмарних послуг. Це призводить до того, що постачальник хмарних послуг може задати параметри інформаційної системи споживача хмарних послуг у відповідності зі сформованою у нього практикою;

– загрози безпеки інформації, пов'язані з втратою довіри. У загальному випадку користувачі хмарних послуг не можуть достовірно оцінити рівень довіри до постачальника хмарних послуг, в зв'язку із закритістю для користувачів застосовуваних постачальником хмарних технологій і програмних рішень. Невідомі еталонні значення будь-яких параметрів, які треба забезпечити постачальнику і користувачеві хмарних послуг для досягнення певного рівня безпеки. Крім того, користувачі хмарних послуг не мають можливість дати оцінку реалізованого рівня захищеності інформації, досягнутого постачальником;

– загрози безпеки інформації, пов'язані з здійсненням несанкціонованого доступу (НСД) з боку споживачів хмарних послуг. Так як хмарні ресурси надаються віддалено, то на хмарний сервер можуть бути проведені атаки, пов'язані з уразливими мережі споживача хмарних послуг;

– загрози безпеки інформації, пов'язані з недоліком управління хмарними ресурсами. Дана загроза зумовлена складністю визначення логічного і фізичного місця розташування хмарних ресурсів, недостатністю фізичного контролю доступу до сховищ даних, надійністю резервного копіювання та ін. Крім того, існують загрози споживачам хмарних послуг, пов'язані з передачею інформації, що захищається постачальникам хмарних послуг, які є резидентами інших країн зі своїми особливостями законодавства в області захисту інформації;

– загрози, пов'язані зі зловживаннями з боку споживачів хмарних послуг. У зв'язку з тим, що споживач хмарних послуг може встановлювати власне ПЗ на хмарний сервер, то для досягнення своїх неправомірних цілей споживач хмарних послуг може встановити шкідливе ПЗ, за допомогою якого здійснювати розсилку спаму, несанкціонованого доступу до віртуальних машин інших клієнтів.

2. Загрози безпеки інформації для постачальників хмарних послуг:

– загрози, пов'язані з невизначеністю при розподілі відповідальності. У хмарі можуть бути визначені такі ролі, як провайдер хмарних послуг, користувач хмарних послуг, адміністратор клієнтської інформаційної системи, власник інформації і т.д. Організація безпеки інформації від таких загроз ускладнена тим, що її реалізація здатна привести до істотних розбіжностей між постачальником і споживачем хмарних послуг з питань, пов'язаних з визначенням їх прав та обов'язків;

– загрози, пов'язані з неузгодженістю політик безпеки. Внаслідок децентралізованого характеру архітектури системи хмарних обчислень, у постачальника і споживача хмарних послуг можуть бути реалізовані різні політики безпеки для одних і тих же засобів захисту;

– загрози, пов'язані з використанням технологій віртуалізації. У більшості випадків основою для створення хмарної інфраструктури є технології віртуалізації. При цьому один фізичний сервер, його обчислювальні ресурси і ресурси пам'яті діляться між безліччю віртуальних машин.

Висновки

В результаті аналізу виявлено такі проблеми забезпечення інформаційної безпеки у хмарних середовищах:

− відсутність в Україні нормативно-правової бази, що регулює питання забезпечення інформаційної безпеки для хмарних обчислень;

− застосування засобів забезпечення інформаційної безпеки, які широко використовуються у сучасних інформаційних системах, знижує ефективність і швидкість обробки інформації замовника хмарних послуг у хмарній інфраструктурі, що зводить на «ні» майже всі переваги хмарних середовищ у порівнянні з традиційними інформаційними системами;

− існуюча система забезпечення інформаційної безпеки хмарного середовища повинна періодично піддаватися незалежному експертному аудиту, який відповідно до вимог міжнародних стандартів, є одним з обов'язкових етапів життєвого циклу будь-якої інформаційної системи.

Література

[1] The NIST Definition of Cloud Computing (англ.). [Електронний ресурс]: Режим доступа: http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspeciallpublication800- 145.pdf

[2] Документ України щодо обробки інформації в системах хмарних обчислень [Электронный ресурс] - Режим доступа : http://w1.c1.rada.gov.ua/pls/zweb2/webproc4_1?pf3511=58527

Проблеми впровадження системи аналітики поведінки користувачів та сутностей

Ігор Пасєка[1], Олександр Сєвєрінов[2]

1. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м Харків, пр. Науки, 14, E-mail: ihor.pasieka@nure.ua

2. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м Харків, пр. Науки, 14, E-mail: oleksandr.sievierinov@nure.ua

Коротка анотація – Issues of implementation of behavior and object analysis systems (UEBA) are considered. The main tasks and components of UEBA systems are considered. The problems that arise when implementing these systems in the organization are analyzed.

Ключові слова - аналітика поведінки користувачів, аналітика поведінки сутностей, UBA, UEBA, внутрішній порушник, інсайдер.

I. Вступ

Останнім часом все більше посилюється небезпека втручання в роботу інформаційних систем в формі несанкціонованого доступу до інформації, в результаті чого кількість витоків конфіденційної інформації з кожним роком зростає. Так за результатами глобального дослідження, проведеного аналітичним центром InfoWatch, у першому півріччі 2019 року зареєстровано 695 витоків інформації, причиною яких став внутрішній порушник (55,6% від загального числа зареєстрованих випадків) та 555 випадків (44,4%) витоку інформації, що стався через зовнішній вплив (рис. 1) [1].

Засоби антивірусного захисту, міжмережні екрани, системи виявлення вторгнень призначені для протидії вірусам і хакерам. Однак вони не захищають від крадіжки інформації власними співробітниками (інсайдерами).

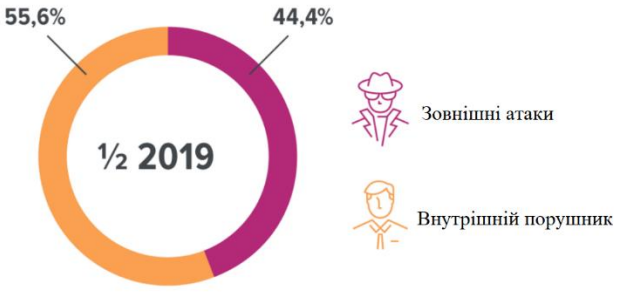


Рисунок 1 – Види порушників при витоку інформації

Одним з найбільш ефективних методів боротьби з несанкціонованим витоком інформації з урахуванням необхідності контролю своїх співробітників є впровадження систем аналітики поведінки користувачів та сутностей – User and Entity Behavior Analytics (UEBA)) [2, 3].

Тому актуальним є проведення аналізу систем UEBA та проблемних питань аналітики поведінки користувачів та сутностей.

II. UEBA - системи аналітики поведінки користувачів та сутностей

Існуючи до недавнього часу на ринку системи інформаційної безпеки, такі як DLP та IAM системи, SIEM рішення, антивірусні та IDS/IPS засоби не можуть у повній мірі вирішити проблеми протидії внутрішнім порушникам. Тому з 2014 року на ринку стали з'являтися системи аналітики поведінки користувачів та сутностей – UEBA. Спочатку вони з'явилися у вигляді модулів для DLP і SIEM систем, мали назву UBA (User Behavior Analytics) та здійснювали тільки аналіз поведінки користувачів. Інші виробники подібних систем називали свої рішення з поведінкового аналізу SUBA (Security User Behavior Analytics) (Forrester), Advanced Analytics або Advanced Threat Analytics (Exabeam і Microsoft) [4].

На даний час системи UEBA являються як окремими рішеннями, так і функціями, що вбудовані в інші платформи безпеки.

Використання UEBA-систем дозволяє перейти до пошуку аномалій не тільки в поведінці користувачів, але і сутностей, до яких відносять робочі станції, мережне обладнання, програмне забезпечення, мережний трафік та інше. Ці рішення дозволяють захиститися від самих різних загроз, таких як несанкціонований доступ та виток конфіденційної інформації, шахрайські дії, крадіжки прав доступу, дії вірусів та шкідливого ПЗ і багато інших.

Враховуючи величезну кількістю даних, що постійно генеруються в інформаційній системі, обробляти їх в ручному режимі неможливо. Тому UEBA-системи набирають популярність.

На даний час системи UEBA вирішують основні завдання [5]:

- аналітика даних з різних джерел (статистична або з використанням методів машинного навчання) в реального часу і/або періодично;
- ідентифікація атак і інших порушень та витоків інформації;
- оперативна пріоритизація подій, отриманих з різних джерел, та видача їх адміністратору інформаційної безпеки;
- надання адміністраторам інформаційної безпеки розширеної інформації про інциденти, яка включає всі об'єкти, що були залучені в аномальну активність.

Дослідницька компанія Gartner розглядає системи UEBA як поєднання трьох компонентів, які включають задачі, що вирішуються, аналітику і джерела даних (рис. 2).

Виходячи з проведених досліджень компанією Gartner, UEBA рішення можуть вирішувати широкий спектр завдань [5]. Але основне застосування даних систем включає виявлення різних категорій загроз при аналізі поведінки користувачів і сутностей:

- неавторизований доступ і переміщення даних;
- підозрілу поведінку привілейованих користувачів;
- шкідливі або неавторизовані дії співробітників;
- нестандартний доступ і використання хмарних ресурсів.

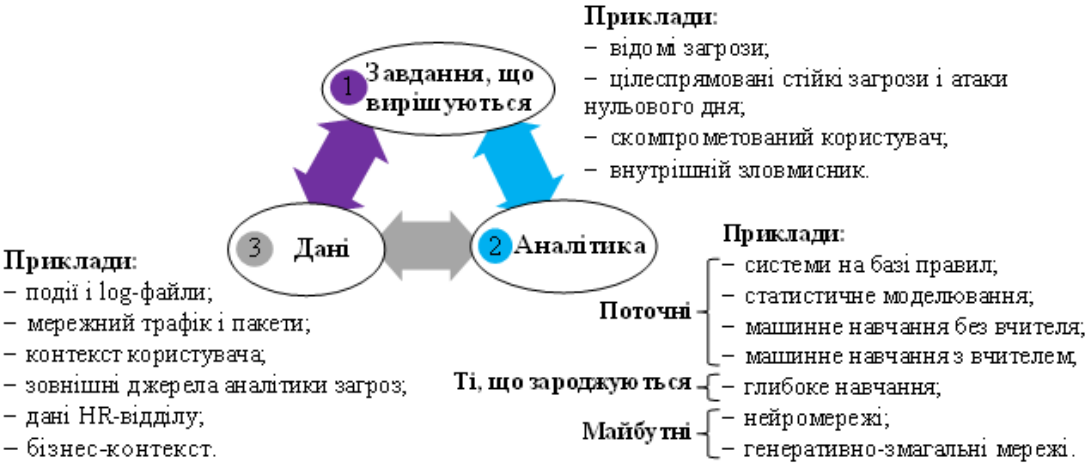


Рисунок 2 – Компоненти системи UEBA

III. Проблеми впровадження UEBA рішень

Проведений аналіз показав, що одними з основних проблем широкого застосування UEBA рішень є їх висока вартість, складне впровадження, обслуговування і використання. Розмір вкладень часу і ресурсів в інструменти UEBA залежить від поставлених завдань і типів аналітики, які необхідні для їх вирішення, і найчастіше вимагають великих витрат. Також, як показує аналіз відгуків користувачів даних систем, час впровадження і повсякденне використання UEBA рішень може бути більш трудомістким і віднімати більше часу, ніж обіцяє виробник, навіть якщо розглядати тільки базові моделі виявлення загроз [5]. Так потрібно від 3 до 6 місяців для запуску UEBA системи з нуля до отримання перших результатів вирішення стандартних завдань. Для більш складних завдань, таких як виявлення інсайдерських погроз в організації, термін може збільшуватися до 18 місяців.

При цьому на ще більше ускладнюють можливість впровадження і ефективність використання систем UEBA такі фактори:

- складність архітектури організації, інформаційної системи та протоколів управління даними;
- доступність даних, потрібних для аналітики UEBA;
- складність алгоритмів аналітики UEBA від виробника та кількість попередньо настроєної аналітики;
- наскільки просто інтегрувати UEBA рішення в вже наявні в організації системи безпеки й аналітики.

На основі проведених досліджень компанія Gartner, враховуючи перелічені проблеми, робить прогноз майбутнього застосування систем аналітики поведінки користувачів та сутностей [5]:

- до 2020 року до 95% всіх впроваджених UEBA-рішень буде складовою частиною інших платформ безпеки (CASB, DLP, EDR, NTA, SIEM);
- до 2021 року ринок систем UEBA як самостійних продуктів, припинить своє існування і буде існувати тільки як інші рішення безпеки з функціоналом UEBA.

Висновки

Таким чином, незважаючи на існуючу потребу в інструментах системи аналітики поведінки користувачів та сутностей та враховуючи існуючі проблеми впровадження UEBA рішень, найбільш імовірно використання в майбутньому функціонала UEBA тільки як складовою частини інших платформ безпеки.

Література

[1] Глобальное исследование утечек конфиденциальной информации в первом полугодии 2019 года. Сайт компанії InfoWatch [Електронний ресурс]. – Режим доступу: https://infowatch.com.

[2] Johansen G. Digital forensics and incident response: an intelligent way to respond to attacks. – 2017.

[3] Сєвєрінов О.В. Аналіз систем аналітики поведінки користувачів і сутностей / О.В. Севєрінов, І.В. Пасєка // Інформатика, управління та штучний інтелект. Тези шостої міжнародної науково-технічної конференції – Х.: НТУ «ХПІ», 2019. – С. 108.

[4] Обзор рынка систем поведенческого анализа — User and Entity Behavioral Analytics (UBA/UEBA) [Електронний ресурс]. – Режим доступу: https://www.antimalware.ru/analytics/Market_Analysis/user-andentity-behavioral-analytics-ubaueba

[5] Сайт компанії Gartner [Електронний ресурс]. – Режим доступу: https://www.gartner.com/.

Вплив вразливостей на функціональні послуги безпеки КСЗІ

В.О. Поддубний[1], В.І. Заболотний[2], А.О. Бойко[3]

1. Кафедра Безпеки інформаційних технологій Харківський національний університет радіоелектроніки, Україна, м. Харків, пр. Науки, 14, E-mail: vadym.poddubniy@nure.ua

2. Кафедра Безпеки інформаційних технологій Харківський національний університет радіоелектроніки, Україна, м. Харків, пр. Науки, 14, E-mail: volodymyr.zabolotnyi@nure.ua

3. Приватне акціонерне товариство "Інститут інформаційних технологій", Україна м.Харків, вул. Бакуліна, 12, E-mail: boyko@iit.kharkov.ua

Коротка анотація –While creating IT system, it is not impossible to make mistakes, sometimes they can be almost useless, but some of them can be used to attack the software or the system. An exploit is a piece of software, a chunk of data, or a sequence of commands that takes advantage of a bug or vulnerability to cause unintended or unanticipated behavior to occur on computer software, hardware. Such behavior frequently includes things like gaining control of a computer system, allowing privilege escalation, or a denial-of-service (DoS or related DDoS) attack.

Nowadays there are several standards of information security management, such as ISO ISO/IEC 27000 series standards.

The purpose of this work is to offer a model of the relationship between vulnerabilities and security services implemented in information and telecommunication systems. Such a system would allow assessing the impact of each of the vulnerabilities on each of the security services implemented in information and telecommunication systems with a comprehensive information security system. Such a model is needed to assess the risks to existing systems and to mitigate the transition to international standards.

Ключові слова – CVSS, ISO/IEC 27000, вразливості, системи управління вразливостями, функціональні послуги безпеки

I Вступ

В сучасному світі неможливо обійтись без стандартизації, яка погоджує різні сфери діяльності, в тому числі і діяльність в галузі інформаційної безпеки. Україна обрала курс на введення міжнародних норм та правил на заміну національним стандартам. Наказ № 631 від 28.12.2010 «Про затвердження національних стандартів України та скасування чинності нормативних документів» [1] вводить до використання цілу низку міжнародних стандартів в різних галузях, в тому числі вперше введений стандарт ДСТУ ISO/IEC 27001:2010 «Інформаційні технології. Методи та засоби досягнення інформаційної безпеки. Системи керування інформаційною безпекою. Вимоги», в подальшому введені і інші стандарти цієї серії в наказах № 1494 від 30.12.2014 [2]; № 631 від 28.12.2010 [3]; № 193 від 18.12.2015 [4]; № 207 від 04.08.2017 [5].

Одним з процесів захисту інформації є процес оцінки ризиків при експлуатації, що передбачає оцінку ризиків від виявлених вразливостей. Зараз відсутні моделі та засоби оцінки впливу виявлених вразливостей на ІТС з КСЗІ. Тому необхідно створити власну модель оцінки вразливостей та її впливу на ІТС з КСЗІ. Така модель необхідна для оцінки ризиків для вже існуючих систем, та для пом'якшення переходу на міжнародні стандарти, оскільки процес переходу від національних до міжнародних стандартів та їх гармонізація займе деякий час.

II Вразливості

Під час розробки ПО виникають помилки, деякі помилки не несуть в собі небезпеки, а деякі становлять серйозну загрозу ІТЗ. Для усунення вразливостей виконується оновлення ПЗ. Однак оновлення може призвести до порушення працездатності ІТС. Так наприклад оновлення Windows 10 October 2018 містило помилки які призводили до видалення файлів користувачів , та та помилки роботи з архівами які могли призвести до втрати даних [6]. А оновлення KB4489894 для Windows 10 могло призводити до аварійного завершення роботи системи [7]. Тому необхідно пріоритезувати встановлення оновлень. Показником для встановлення пріорітета є співвідношення потенціальних ризиків від неусунення вразливостей та ризиків від встановлення оновлень.

Але для всіх систем управління ризиками необхідна база вразливостей, яка буде містити інформацію про вплив вразливості на ІТС. Для оцінки впливу вразливості зазвичай використовують систему оцінки вразливостей CVSS версії 2.0 або 3.0. За допомогою даної системи можна слідкувати за вразливостями, та приймати рішення що до мір реагування. CVSS намагається призначити показники критичності вразливості, що дозволяє респондентам визначати пріоритети дій та ресурсів відповідно до загрози. Оцінки розраховуються на основі формули, яка залежить від кількох показників. Оцінки коливаються від 0 до 10, де оцінка 10 означає найвищий ступінь критичності вразливості. Рівень небезпеки можна оцінити за шкалою **FortiGuard,** де оцінка за системою CVSS переводиться в рівні критичності.Оцінка від 0.1 до 3.9 значить низький рівень загрози, від 4 до 6.9 середній, від 7 до 8.9 високий, від 9 до 10 критичний. Так наприклад за період з 07.10.2019 по 13.10.2019 було виявлено 253 нові вразливості, 48,6% низької загрози, 26,1 середньої, 25,3 високої. При цьому 16,2% вразливостей не було виправлено на момент написання роботи [8].

III Функціональні послуги безпеки

В Україні на даний момент діє Нормативний документ НД ТЗІ 2.5-004-99 «Критерії оцінки захищеності інформації в комп'ютерних системах від

несанкціонованого доступу». Цей документ є методологічною базою для визначення вимог з захисту інформації в комп'ютерних системах від несанкціонованого доступу; створення захищених комп'ютерних систем і засобів захисту від несанкціонованого доступу; оцінки захищеності інформації в комп'ютерних системах і їх придатності для обробки критичної інформації (інформації, що вимагає захисту).

В процесі оцінки спроможності комп'ютерної системи забезпечувати захист оброблюваної інформації від несанкціонованого доступу розглядаються вимоги двох видів:

1) вимоги до функцій захисту (послуг безпеки);

2) вимоги до гарантій.

В контексті документу комп'ютерна система розглядається як набір функціональних послуг. Кожна послуга являє собою набір функцій, що дозволяють протистояти певній множині загроз. Кожна послуга може включати декілька рівнів. Чим вище рівень послуги, тим більш повно забезпечується захист від певного виду загроз. Рівні послуг мають ієрархію за повнотою захисту, проте не обов'язково являють собою точну підмножину один одного.

Функціональні послуги згруповані за властивостями інформації та ІТС. Кожна з властивостей (конфіденційність, цілісність, доступність, спостережність).

Послуги є взаємозв’язаною множиною, наприклад рівень послуги «Цілісність комплексу засобів захисту» НЦ-1 є необхідною умовою абсолютно для всіх рівнів всіх інших послуг [9].

IV Вплив вразливостей на функціональні послуги безпеки

Знайдена вразливість може вплинути на реалізацію однієї, або декількох послуг , або не мати впливу ІТС зовсім. Наприклад вразливість призводить до порушення конфіденційності, але у ІТС не висувається вимог до конфіденційності. Тому вразливість не є критичною. Або навпаки сама вразливість може бути не критичною, але вплив який вона спричиняє на низку взаємозв'язаних послуг є фатальним, така ситуація потребує негайного рішення. Враховуючи, властивості вразливостей необхідно знати як вплине наявність вразливості на кожну конкретну ІТС з КСЗІ, щоб підрахувати ризики пов’язані з наявною вразливістю в системи, та прийняти рішення що до усунення даної вразливості або продовження роботи ІТС незважаючи на неї. Тому необхідно розробити систему яка б відображала вплив властивостей вразливості на реалізовані функціональні послуги безпеки.

Висновок

На даний момент актуальною задачею є розробка моделі взаємозв’язку між вразливостями та послугами безпеки які реалізуються в ІТС. Така система дозволила б оцінити вплив кожної з вразливостей на кожну з послуг безпеки, які реалізуються в ІТС з КСЗІ. Така модель необхідна для оцінки ризиків для вже існуючих систем, та для пом’якшення переходу на міжнародні стандарти.

Література

[1] Наказ №631 від 28.12.2010 «Про затвердження національних стандартів України та скасування чинності нормативних документів»[Електронний ресурс] – Режим доступу: https://zakon.rada.gov.ua/rada/show/v0631831-10

[2] Наказ №1494 від 30.12.2014 «Про прийняття європейських та міжнародних нормативних документів як національних стандартів України, змін до національних стандартів України, скасування національних стандартів України та міждержавних стандартів в Україні» [Електронний ресурс]. – Режим доступу: https://zakon.rada.gov.ua/rada/show/v1494731-14

[3] Наказ № 631 від 28.12.2010 «Про затвердження національних стандартів України та скасування чинності нормативних документів» [Електронний ресурс]. – Режим доступу: https://zakon.rada.gov.ua/rada/show/v0631831-10

[4] Наказ № 193 від 18.12.2015 «Про прийняття нормативних документів України, гармонізованих з міжнародними та європейськими нормативними документами, скасування національних стандартів України» [Електронний ресурс]. – Режим доступу: zakon.rada.gov.ua/rada/show/v0193774-15

[5] Наказ № 207 від 04.08.2017 «Про прийняття національних нормативних документів, гармонізованих з європейськими нормативними документами, поправки до національного нормативного документа, скасування національних нормативних документів» [Електронний ресурс]. – Режим доступу: https://zakon.rada.gov.ua/rada/show/v0207774-17

[6] В Windows 10 October 2018 Update обнаружена очередная ошибка [Електронний ресурс]. – Режим доступу: https://www.securitylab.ru/news/496057.php

[7] Мартовское обновление для Windows 10 вызывает «синий экран смерти» [Електронний ресурс]. – Режим доступу: https://www.securitylab.ru/news/498417.php

[8] Vulnerability Database [Електронний ресурс] - Режим доступу: https://www.cybersecurity-help.cz/vdb/

[9] НД ТЗІ 2.5-004-99 «Критерії оцінки захищеності інформації в комп’ютерних системах від несанкціонованого доступу»; Департамент спеціальних телекомунікаційних систем та захисту інформації Служби безпеки України, 1999. – 61с.(Нормативний документ)

Оцінка критичності вразливостей в операційних системах

Карина Ревізорова[1], Тетяна Гріненко[2]

1. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м. Харків, пр. Науки, 14, E-mail: karinamaroon71@gmail.com

2. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м. Харків, пр. Науки, 14, E-mail: tetiana.grinenko@nure.ua

Коротка анотація – Analysis of the vulnerability problem in operating systems. Comparison of existing vulnerability criticality assessment techniques. Review of existing software solutions in the global market. Sample metrics to assess vulnerability criticality.

Ключові слова – Вразливість, загроза, оцінка критичності вразливості.

I. Вступ

Одним з важливих заходів захисту інформації в комп'ютерних системах є визначення переліку загроз інформації. Загроза – можлива причина небажаного інциденту, який може завдати шкоди системі або організації [1]. Одна або декілька загроз можуть використовувати ряд вразливостей інформації. Вразливість в інформаційних технологіях (ІТ) – недолік або слабке місце, яке може бути використане для реалізації загрози [2]. Причинами виникнення вразливостей можуть бути помилки при проектуванні та в період розробки програмного (програмно-апаратного) забезпечення, неправомірна зміна режимів роботи пристроїв і програм або збої в їх роботі. Вразливість слід оцінювати, розглядаючи всі загрози, які можуть використовувати її у конкретній ситуації. Наслідками реалізації атаки на інформацію за допомогою вразливостей є: несанкціоноване розкриття, модифікація, пошкодження, знищення, недоступність або втрата інформації [2].

Вразливість, з моменту її виявлення і до випуску патча, має назву «вразливість нульового дня». Програма з такою вразливістю може бути зламана в будь-який момент, і відомості про такі вразливості так само є товаром. Після закриття вразливості (випуску патча) цінність вразливості падає. Вразливість, яка не має відповідної загрози, може не вимагати патчів, але розробники повинні її враховувати і піддавати моніторингу на предмет змін. Слід зазначити, що невірно реалізований або неправильно функціонуючий засіб контролю теж може бути вразливістю [2].

Більшість хакерських атак стають можливими через наявність вразливостей в існуючих операційних системах (ОС). ОС є найважливішим програмним компонентом будь-якої обчислювальної машини, тому від рівня реалізації політики безпеки в кожній конкретній ОС залежить і загальна безпека інформаційної системи [3]. Загрози безпеки ОС істотно залежать від умов експлуатації системи та від інформації, що зберігається і обробляється в системі, тощо. Наприклад, якщо ОС використовується для організації електронного документообігу, найбільш небезпечними є загрози, що пов'язані з порушенням цілісності та доступності інформації [4].

II. Засоби оцінки критичності вразливостей

Існує ряд систем оцінки вразливостей, які створені комерційними та некомерційними організаціями, наприклад, CERT/CC, система аналізу вразливостей SANS, система оцінки від Microsoft, CVSS, тощо [3].

Кожна з систем має свої переваги, але всі вони відрізняються за ознакою вимірювання. Наприклад, CERT/CC використовує значення оцінок від 0 до 180 і враховує такі фактори: чи схильна Інтернет-інфраструктура до ризику та який тип передумов потрібен для експлуатації вразливості. Система аналізу вразливостей SANS враховує: в якій конфігурації знайдена вразливість (стандартній чи ні), чи є система клієнтом або сервером [5]. Система оцінки від Microsoft намагається відобразити складність експлуатації та загальний вплив від експлуатації вразливості. Ці системи оцінки є корисними, але вони представляють підхід, при якому вважається, що наслідки експлуатації вразливості однакові для приватної особи і для компанії [5].

В результаті аналізу ринку програм з аналізу рівня критичності вразливостей, були виділені три продукти:

– PT Exploit Explorer (безкоштовна). Плюси: можливість завантажувати текстові файли; має корисні посилання на експлоіти. Мінуси: джерело для утіліти – база даних вразливостей CVE, яка дає мало інформації для оцінки; застосування тільки для вразливостей, для яких є експлоіти; неявно описано, яким чином здійснюється оцінка критичності вразливостей [5].;

– калькулятор CVSS V2 (безкоштовна). Плюси: детальний огляд метрик, за якими здійснюється оцінка. Мінуси: інтуїтивно незрозумілі критерії відбору; суперечлива логіка оцінювання;

– Nessus Vulnerability Scanner (обмежена free версія). Плюси: містить актуальні моделі загроз, на можливість експлуатації яких сканер швидко перевіряє клієнтську мережу. Мінуси: демонструє критичність вразливостей, тільки знайдених при скануванні [5].

III. Вибір метрик для оцінки критичності вразливостей

Кількісна оцінка безпеки має різні галузі. В залежності від того, як розглядати систему, метрики, що використовуються для оцінки критичності вразливостей можуть відрізнятися. Потрібно обирати метрики, які є практичними, корисними і значущими.

На підставі аналізу були вибрані наступні метрики впливу вразливостей, що дозволяють зловмиснику:

–впливати на конфіденційність (є влив/ не визначено);

–впливати на цілісність (є влив/ не визначено);

–впливати на доступність (є влив/ не визначено);

–дії направлені на пошкодження пам'яті (використовується/ не визначено);

–підбір паролів (використовується/ не визначено);

–спричинення до відмови в обслуговуванні (є влив/ не визначено);

–несанкціонована зміна прав доступу (використовується/ не визначено);

–злонамірне виконання довільного коду (використовується/ не визначено).

Також перевірка на:

–необхідність автентифікації для реалізації атаки (легко/ не визначено);

–складність експлуатації вразливості (легко/ не визначено);

–локальна або зовнішня взаємодія (локальна/зовнішня/ не визначено).

Ці метрики були вибрані на підставі аналізу опису вразливостей у ОС Windows за останні три місяці у БД вразливостей NVD.

IV. Аналіз вибраних метрик та порівняння оцінок

Щоб уникнути великої розбіжності результатів, була вибрана шкала від 0 до 10 та умовні позначення Low, Medium, High та Critical. Для кожного параметра вибраних метрик на основі експертної оцінки та порівняння показників у калькуляторі CVSS були виставлені коефіцієнти, в залежності має чи ні (чи не відомо) вразливість вплив на конфіденційність, цілісність, доступність, пошкодження пам’яті, та ін. Також на результат впливають вагові коефіцієнти складності експлуатації вразливості, чи є необхідність автентифікації для реалізації атаки та тип взаємодії (локальна, зовнішня чи не визначено).

Для порівняння у табл. 1 наведені результати розрахунків, які надає БД NVD, та розрахунки, що отримані на підставі вибраних метрик та вагових коефіцієнтів (у табл. 1 позначається як «*»).

ТАБЛИЦЯ 1

РЕЗУЛЬТАТИ РОЗРАХУНКІВ

Id	CVSSv3	Nvd.nist.gov	*
CVE-2018-4127	8,8	6,8	8,1
CVE-2017-7172	7,8	9,3	9
CVE-2018-6251	7,2	7,8	7,5

На підставі первинного аналізу (більше 100 вразливостей), відмінність між показниками CVSSv3 та Nvd.nist.gov становить 12,6%, а відмінність між CVSSv3 та «*» – 10,1% та між Nvd.nist.gov та «*» – лише 2,5%, можна зробити висновок, що параметри та коефіцієнти для оцінки були вибрані оптимально.

Висновок

Оцінка критичності вразливостей складний і суперечний процес, який складається з великої системи метрик, параметрів та експертних оцінок. Проведений аналіз методик оцінки критичності вразливостей існуючих програмних рішень на світовому ринку показав, що на даний момент не існує програмного забезпечення, яке могло б: аналізувати вразливості без допоміжних втручань користувача; аналізувати не тільки вразливості, для яких існують експлоіти; демонструвати вибірку вразливостей для кількох ОС у зручному форматі; проводити інтеграцію з базою даних вразливостей NVD (більш інформативною ніж CVE).

Резюмуючи вказане у другому розділі, обґрунтовано вимоги до методики оцінки вразливості:

–можливість оцінки ступеня ризику вразливості в залежності від можливості її експлуатації;

–результатом застосування методики має бути числове значення, що підходить для використання при аналізі ризиків;

–методика повинна мати можливість адаптації до конкретної інформаційної системи;

–параметри, які використовуються при розрахунку, повинні допускати мінімум різночитань;

–механізм розрахунку результуючого значення має бути простий і зрозумілий.

Враховуючи ці критерії, була розроблена методика, яка була взята за основу для розробки програмного продукту, що дозволить більш якісно і інформативно проводити оцінку критичності вразливостей.

Література

[1] ISO/IEC 27000 Информационные технологии - Методы и средства обеспечения безопасности - Системы менеджмента информационной безопасности – Общие сведения и словарь [Електроний ресурс] // ISO/IEC – 20.10.2019. Режим доступу: http://pqm-online.com/assets/files/pubs/translations/std/iso-mek-27000-2016.pdf.

[2] Модель угроз ПД. Организационно-распорядительная документация по защите ПД [Електроний ресурс] // Институт.ру – Режим доступу: http://www.intuit.ru/studies/courses/ 697/553/lecture/12447 = 20.10.2019.

[3] Безбогов А.А. Безопасность операционных систем : учебное пособие / А.А. Безбогов, А.В. Яковлев, Ю.Ф. Мартемьянов. – М.:"Издательство Машиностроение-1", 2007. – 220 с.

[4] Проблемы обеспечения безопасности ОС [Електроний ресурс] // Your Private Network. Режим доступу: http://ypn.ru/301/operating-systems-security-problems/- 31.10.2019.

[5] Полное руководство по общему стандарту оценки уязвимостей [Електроний ресурс] // securitylab. Режим доступу: https://www.securitylab.ru/ analytics/355336.php; - 31.10.2019.

Порівняльний аналіз систем виявлення і запобігання вторгнень

Олександр Риков[1], Інна Олешко[1]

1. Кафедра безпеки інформаційних технологій,
Харківський національний університет радіоелектроніки,
УКРАЇНА, м. Харків, пр. Науки, 14,
E-mail: oleksandr.rykov@nure.ua

Коротка анотація – The article is written about multitasking protecting program that helps users detecting intruders. It deals with application and shows how people can use this app to prevent stealing of their data. It was shown, that the Suricata program is one of the most popular and fast protector.

Ключові слова - адміністратор, трафік, мережева атака, Suricata, IDS, IPS, Snort.

I. Вступ

У сучасному світі системи виявлення та запобігання вторгнень (Intrusion detection system / Intrusion prevention system, IDS / IPS) - необхідний елемент захисту від мережевих атак. Основне завдання даних систем - виявлення фактів несанкціонованого доступу в корпоративну мережу або несанкціонованого управління нею, з виконанням відповідних заходів протидії (інформування адміністраторів про факт вторгнення, обрив з'єднання або перенастроювання брандмауера для блокування подальших дій зловмисника і т.д.).

Існує багато систем виявлення та запобігання вторгнень. Актуальною є задача вибору однієї з них. У роботі проводиться порівняльний аналіз систем виявлення вторгнень, робиться висновок про те, що система Suricata є більш швидким та надійним детектором атак.

II. IDS / IPS-рішення

IDS / IPS системи - це унікальні інструменти, створені для захисту мереж від несанкціонованого доступу. Вони являють собою апаратні або комп'ютерні засоби, які здатні оперативно виявляти і ефективно запобігати вторгненням. Серед заходів, які приймаються для досягнення ключових цілей IDS / IPS, можна виділити інформування фахівців з інформаційної безпеки про факти спроб хакерських атак і впровадження шкідливих програм, обрив з'єднання зі зловмисниками і перенастроювання мережевого екрану для блокування доступу до корпоративних даних.

Кібератаки - одна з основних проблем, з якими стикаються суб'єкти, які мають інформаційними ресурсами. Відомі антивірусні програми і брандмауери ефективні лише для захисту очевидних місць доступу до мереж. Однак зловмисники здатні знаходити шляхи обходу і вразливі сервіси навіть в найдосконаліших системах безпеки. При такій небезпеці не дивно, що закордонні та українські UTM-рішення отримують все ширшу популярність серед організацій, що бажають виключити можливість вторгнення і поширення шкідливого ПЗ (хробаків, троянів і комп'ютерних вірусів). Багато компаній приймають рішення купити сертифікований міжмережевий екран або інший інструмент для комплексного захисту інформації.

Всі існуючі сьогодні системи виявлення та запобігання вторгнень об'єднані кількома загальними властивостями, функціями і завданнями, які з їх допомогою вирішують фахівці з інформаційної безпеки. Такі інструменти за фактом здійснюють безперервний аналіз експлуатації певних ресурсів і виявляють будь-які ознаки нетипових подій.

Організація безпеки корпоративних мереж може ґрунтуватися на кількох технологіях, які відрізняються типами виявлених інцидентів і методами. Крім функцій постійного моніторингу та аналізу того, що відбувається, IDS системи виконують такі функції:

- збір і запис інформації;
- оповіщення адміністраторів мереж про зміни, що відбулися;
- створення звітів для підсумовування логів.
- Технологія IPS в свою чергу у додачу до вище сказаного, здатна не тільки визначити загрозу і її джерело, а й здійснити їх блокування. Це говорить про розширений функціонал подібного рішення. Вона здатна здійснювати наступні дії:
- обривати шкідливі сесії і запобігати доступу до найважливіших ресурсів;
- змінювати конфігурацію «підзахисної» середовища;
- проводити дії над інструментами атаки (наприклад, видаляти заражені файли).

Варто відзначити, що UTM-міжмережевий екран і будь-які сучасні системи виявлення та запобігання вторгнень є оптимальною комбінацію технологій систем IDS і IPS.

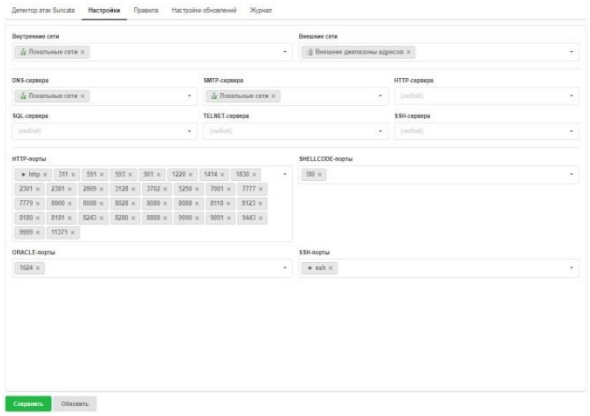
III. Детектори атак Suricata

Одним з рішень IPS запобігання вторгнень є детектори атак, які призначені для своєчасного виявлення безлічі шкідливих загроз. В Інтернет Контроль Сервері вони реалізовані у вигляді системи Suricata - багатозадачного і продуктивного інструменту, розробленого для захисту мереж, а також збору і зберігання інформації про будь-які сигнали, що надходять. Робота детектора атак заснована на аналізі сигнатур і евристиці, а зручність його обумовлено наявністю відкритого доступу до вихідного коду. Такий підхід дозволяє налаштовувати параметри роботи системи для вирішення індивідуальних завдань.

До редагованих параметрів Suricata відносяться правила, яким буде підпорядковуватися аналіз трафіку, фільтри, що обмежують висновок попередження адміністратора, діапазони адрес різних серверів, активні порти і мережі.

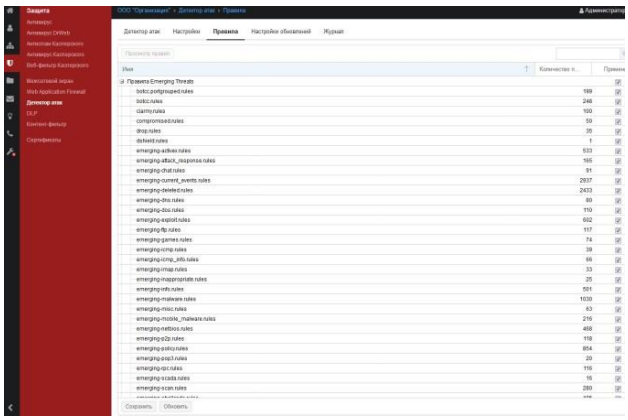
Таким чином, Suricata, як IPS-рішення - це досить гнучкий інструмент, функціонування якого підлягає змінам в залежності від характеру атаки, що робить його максимально ефективним.

В ІКС фіксується і зберігається інформація про підозрілу активність, блокуються ботнети, DOS-атаки, а також TOR, анонімайзери, P2P і торрент-клієнти.



Малюнок 1 – Настройка “Suricata”

У вкладці налаштувань (малюнок 1) можна редагувати параметри роботи детектора атак. Тут можна вказати внутрішні, зовнішні мережі, діапазони адрес різних серверів, а також порти, що використовуються. Всім цим змінним присвоєно значення за замовчуванням, з якими детектор атак може коректно запуститися. За замовчуванням, аналізується трафік на зовнішні інтерфейси.



Малюнок 2 – Правила “Suricata”

Детектору атак можна підключати правила, за допомогою яких він буде аналізувати трафік. На вкладці на малюнку 2 можна подивитися наявність і вміст того чи іншого файлу з правилами, а також включити або виключити його дію (за допомогою прапорців праворуч). У правому верхньому куті розташовується пошук за назвою або за кількістю правил у файлі.

IV. Порівняння Suricata та Snort

Більше 250 одиничних тестів було проведено проти Suricata та Snort, дотримуючись методології, результати наведені в Таблиці 1.

ТАБЛИЦА 1

РЕЗУЛЬТАТИ ТЕСТІВ

Тестова група	Кількість тестів	Оцінка suricata	Оцінка snort
Поганий трафік	4	1	1
Роздроблені пакети	2	1	3
Шкідливі програми та віруси	14	9	7
Відмова в обслуговуванні (DoS)	3	3	3
Атаки з боку клієнта	257	157	127
Оболонки	12	12	7
Продуктивність	0	2	1
Всього	297	185	149

Випробування були проведені на 14 шкідливих програмах та вірусах. Suricata має кращий рівень виявлення шкідливих програм та вірусів, ніж Snort.

На наборі з 11 оболонок (вірус схований в іншому файлі), Suricata виявив 9 shellcodes, а Snort виявив 7 shellcodes.

У наборі з 3 тестів і Suricata, і Snort виявили 3 спроби DoS проти служб SSH та MSSQL.

Тести продемонстрували, що Suricata краща, ніж Snort для виявлення нападів на стороні клієнта, зі швидкістю виявлення 82% проти 49%.

Висновки

За результатами наведеного вище аналізу робимо висновок про те, що детектор атак Suricata – швидка та досить надійна систем, яка вміє по максимуму використовувати можливості сучасних процесорів і GPU. Мінус цієї системи - велика кількість налаштувань і недостатньо виразна в деяких питаннях документація. Проте, після установки, система досить добре працює з настройками за замовчуванням, а з тонким налаштуванням досвідчений адміністратор безпеки розбереться без проблем.

Література

[1] Anderson, James P., "Computer Security Threat Monitoring and Surveillance, " Washing, PA, James P. Anderson Co., 1980.

[2] Електронний ресурс https://oisf.net (дата звернення 20.10.2019)

[3] Електронний ресурс https://suricata-ids.org (дата звернення 20.10.2019)

Ідентифікація користувача за його поведінкою в системі

Ігор Рубан[1], Віталій Улітічев[2]

1. Кафедра електронних обчислювальних машин,
Харківський національний університет радіоелектроніки,
УКРАЇНА, г. Харків, пр. Науки, 14,
E-mail: ihor.ruban@nure.ua

2. Кафедра безпеки інформаційних технологій,
Харківський національний університет радіоелектроніки,
УКРАЇНА, г. Харків, пр. Науки, 14,
E-mail: vitalii.ulitichev@nure.ua

Коротка анотація – The widespread information systems that integrate client server technology with the global network have posed numerous problems. It turned out that standard identification methods are already out of date. Particularly the problem is that the universally recognized separation of methods of controlling access to and access to information is no longer effective. To resolve this issue, you need to provide identification methods that will be able to identify users through a set of actions taken by users while working with DIS.

Ключові слова – інформаційні системи, нейронна мережа, автентифікація, ідентифікація.

I. Вступ

Ідентифікація та автентифікація можуть вважатися основними програмними та технічними засобами безпеки, оскільки решта служб орієнтована на підтримку названих об'єктів. Ідентифікація та автентифікація - це перша лінія захисту інформаційної зони організації.

Відповідно до звіту [1] інституту ДАНС «Тенденції витрачання на IT-безпеку», який вказує на відсоток витрат організацій на забезпечення різних технічних засобів інформаційної безпеки. На рис 1 представлена схема вартості кожної технологічної одиниці.

Проаналізувавши діаграму, можна визначити два типи технологій кібербезпеки, які представлені найважливішими, такими як:

– доступ до автентифікації;

– розширити захист від зловмисного програмного забезпечення;

Зараз розповсюджені інформаційні системи стають популярнішими, ніж раніше. Сумнівно, що існує можливість знайти будь-яку програму, яка не використовує компоненти різних постачальників. Чим складніше сучасні програми, тим більше потребує використання компонентів, розподілених віддаленою машиною. Послуги, які використовуються веб-додатками, дуже часто розміщуються не на території інформаційного центру закладу.

Поширені інформаційні системи, що об'єднують технологію сервера клієнта з глобальною мережею Інтернет, викликають численні проблеми. Виявилося, що стандартні методи ідентифікації вже застаріли. Особливо проблема полягає в тому, що загальновизнаний поділ методів контролю за фізичним доступом і контролю над доступом до інформації вже не дієвий. Для вирішення цієї проблеми необхідно застосувати методи ідентифікації, які матимуть можливість ідентифікувати користувачів за допомогою сукупності дій, реалізованих користувачами в процесі роботи з DIS.

II. Основний матеріал

Найпоширенішими сьогодні є методи ідентифікації користувачів, засновані на використанні паролів, які, на жаль, можуть бути втрачені, викрадені або порушені багатьма способами. Таким чином, може бути чудово реалізувати ідею комбінації стандартного пароля з методом ідентифікації користувачів на основі їх поведінки в системі. У цьому випадку, навіть якщо зловмисник отримує доступ до пароля, загальний доступ до комп'ютерної системи може бути відхилений через ідентифікацію поведінки, зображену на. У цьому випадку структура ідентифікації через поведінку користувача в системі є наступне: якщо користувач введе неправильний пароль, йому буде відмовлено у доступі негайно. Якщо пароль є

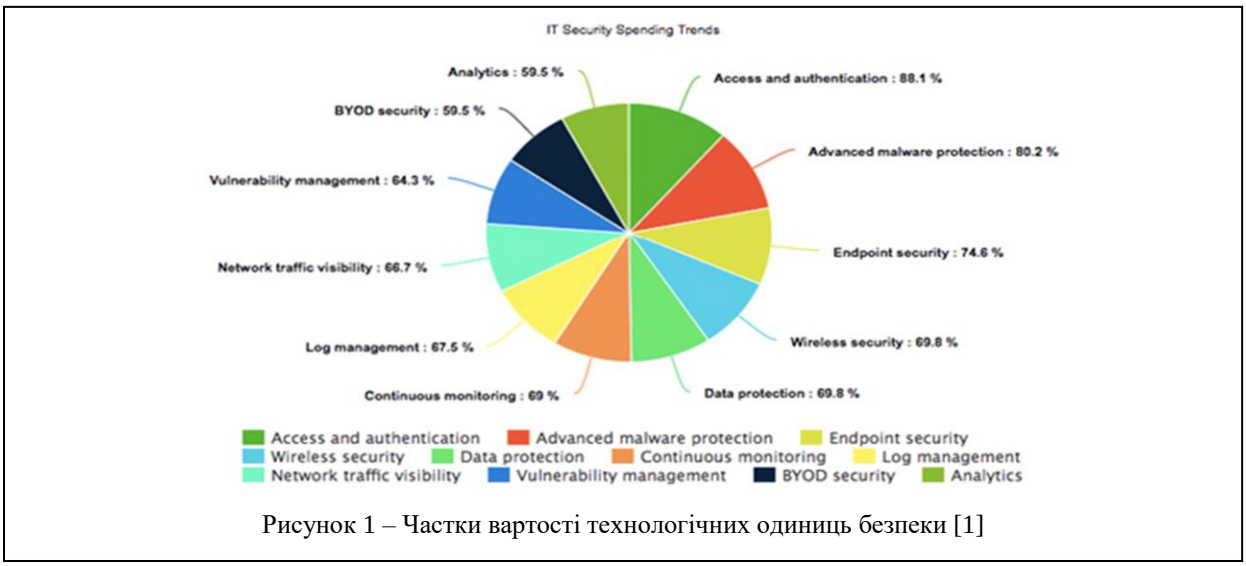


Рисунок 1 – Частки вартості технологічних одиниць безпеки [1]

правильним протягом певного періоду часу, T відбувається процес збору даних про його поведінку, після чого ці дані порівнюються з прикладом зареєстрованого користувача з бази даних, яка вже пройшла через ідентифікацію. Залежно від необхідної точності в процесі даних порівняння доступу може бути дозволено або заборонено.

Завдання ідентифікації можна розділити на кілька етапів, основними з яких є процес викладання та, відповідно, процес розпізнавання. На початку система бере і зберігає в базі даних інформацію про поведінку користувача в системі; на основі значень цих налаштувань існує шаблон або профіль поведінки користувача. Потім відбувається процес порівняння даного шаблону з уже збереженим у базі даних в системі, який є фактично ідентифікаційним.

Ідентифікація відбувається завдяки результатам контролю за поведінкою користувача в процесі роботи в інформаційній системі. Як вихідні дані використовуються матричні індикатори X у доказуванні моніторингу поведінки користувача та стовпець Y, який складається з множинності {0,1}, де 1 виникає, якщо поведінка користувача стосується визначеного користувача та 0, якщо ні.

Матриця X складається з векторів, які мають таку форму:

$$x_i = \{S_i^T, ST_i^T\}, i = 1, \dots, l$$

де S_i^T – множина станів в яких знаходився користувач за період і-ої сесії, а ST_i^T – множина статистичних параметрів побудованих на основі множини S_i^T, наприклад тривалість перебування користувача в кожному стані з множини S_i^T, середній час перебування користувача в кожному стані S_i^T та інші. Сесії користувачів виділені таким чином, що вони не можуть бути довше часу T або містить більш B станів. Тобто сесія вважається закінченою або коли користувач знаходився в B станах, або коли сесія зайняла за часом більш T часу .

Завдання побудови шаблону поведінки користувача буде складатися в побудові функції

$$\alpha: X \to Y, \qquad (2)$$

Яка здатна ідентифікувати довільний об'єкт $x_i \in X$. Для побудови функції шаблону α можна використовувати лінійну модель з вектором параметрів $w=\{w_0, w_{1,\dots,} w_n\}$, де n довжина вектора x_i. Звідси функція шаблон α має такий вигляд:

$$\alpha(x, w) = w_0 + w_1 x_1 + \dots + w_n x_n$$

при цьому завдання ідентифікації користувача можна звести до задачі бінарної класифікації з множиною $Y=\{-1;1\}$. В цьому випадку шаблон користувача буде мати наступний вигляд:

$$\alpha(x, w) = sign \sum_j^n w_j x_j, \ (x_0 = 0)$$

Знайдені параметри і будуть шаблоном поведінки користувача і забезпечувати оптимальне значення функціоналу якості. У цьому завданню мінімізується функціонал помилок - це середня кількість розбіжностей, де $L(\alpha, x_i)$ – це функція втрат.

Функція помилок має такий вигляд:

$$Q(\alpha, X) = \frac{1}{l}\sum_i^l L(a, x_i) = \frac{1}{l}\sum_i^l [a(x_i) - y_i] \to min, \quad (5)$$

Висновки

Підвівши підсумки результатів експерименту можна виділити наступні переваги і недоліки ідентифікації користувача по його поведінці в інформаційній системі.

Переваги.

Простота реалізації і впровадження. Не потрібно спеціального апаратного забезпечення, дані беруться з системи моніторингу стану інформаційної системи, а значить - використання не потрібне придбання ніякого додаткового обладнання. Це найдешевший спосіб ідентифікації по біометричних характеристик суб'єкта доступу.

Не вимагає від користувача ніяких додаткових дій і навичок. не представляється можливим скопіювати профіль поведінки аутентифицированного користувача.

- Можливість прихованої ідентифікації.

Недоліки.

На етапі первинного експлуатування потрібні додаткові витрати по часу, для побудови профілю поведінки.

Сильно залежить від безлічі станів в яких може перебувати конкретний користувач системи.

Список літератури

[1] [SANS 2016м IT Security Spending Strategies Survey // [Online] https://www.sans.org/webcasts/2016-security-spending-strategies-survey-100997

[2] How to Get the Absolute Most from Your Cybersecurity Budget // [Online] https://www.stickman.com.au/how-to-get-the-absolute-most-from-your-cybersecurity-budget/

[3] Kim H., Lee E. A. Authentication and Authorization for the Internet of Things //IT Professional. – 2017. – Т. 19. – №. 5. – С. 27-33.

[4] Ali M. L. et al. Keystroke biometric systems for user authentication //Journal of Signal Processing Systems. – 2017. – Т. 86. – №. 2-3. – С. 175-190.

[5] Wu F. et al. A lightweight and robust two-factor authentication scheme for personalized healthcare systems using wireless medical sensor

Захист веб-додатків від SQL-ін’єкцій

Гліб Саєнко[1], Тетяна Гріненко[2]

1. Кафедра безпеки інформаційних технологій,
Харківський національний університет радіоелектроніки,
м. Харків, пр. Науки, 14,
E-mail: saenkogleb21@gmail.com

2. Кафедра безпеки інформаційних технологій,
Харківський національний університет радіоелектроніки,
м. Харків, пр. Науки, 14,
E-mail: tetiana.grinenko@nure.ua

The report examined the main types of SQL-injection and methods of protection against them.

Атака, база даних, веб-додаток, вразливість, SQL-ін’єкція.

I. Вступ

Потреби ринку інформаційних послуг нарівні з бурхливим розвитком телекомунікаційних технологій в останні десятиріччя стали вагомими причинами стрімкого розвитку технологій широкого обміну інформацією, побудови потужних територіальних і глобальних обчислювальних мереж. Такою мережею є Інтернет і найбільшу частину у ньому займають веб-додатки (веб-сайти). Існує величезна кількість різного роду вразливостей, за допомогою яких зловмисник може зламати веб-додаток і видалити або вкрасти чужу інформацію. Однією з таких уразливостей є SQL-ін’єкція – впровадження шкідливого SQL-коду в тіло HTTP-запиту до веб-додатку [1].

II. SQL-ІН’ЄКЦІЯ

SQL-ін’єкція – це один з поширених способів злому сайтів та програм, що працюють з базами даних, заснований на впровадженні в запит до веб-додатку довільного SQL-коду [1,2].

Атака типу впровадження SQL-коду, в залежності від типу системи управління базами даних та умов впровадження, дає можливість атакуючому виконати довільний запит до бази даних (наприклад, прочитати вміст будь-яких таблиць, видалити, змінити або додати дані), отримати можливість читання та/або запису локальних файлів та виконання довільних команд на сервері. Атака типу впровадження SQL-коду може бути можлива за некоректної обробки вхідних даних, що використовуються в SQL-запитах. Для безпеки веб-додатків рекомендується, щоб розробники ніколи не використовували введені користувачем дані безпосередньо для формування SQL-запитів до бази даних. Найкращим рішенням є, звичайно, повна ізоляція веб-додатків від створення SQL-запитів, тобто перенесення всього функціоналу, пов'язаного з їх формуванням, на рівень збережених процедур серверу баз даних.

III. Приклади SQL-ІН’ЄКЦІЙ

Припустимо, що серверне програмне забезпечення, отримавши вхідний параметр id, використовує його для створення SQL-запиту. Розглянемо такий PHP-скрипт [2]:

```
$id = $_REQUEST['id'];
$res = mysql_query("SELECT * FROM news WHERE id_news = $id").
```

Якщо на сервер переданий параметр id, що дорівнює 5 (наприклад: http://example.org/script.php?id=5), то буде виконаний такий SQL-запит:

```
SELECT * FROM news WHERE id_news = 5.
```

Але, якщо зловмисник передасть як параметр id рядок -1 OR 1=1 (наприклад, таким чином: http://example.org/script.php?id=-1+OR+1=1), то виконається запит:

```
SELECT * FROM news WHERE id_news =-1 OR 1=1.
```

Таким чином, зміна вхідних параметрів шляхом додавання в них конструкцій мови SQL викликає зміну в логіці виконання SQL-запиту (в цьому прикладі замість новини із заданим ідентифікатором будуть вибрані всі наявні в базі новини, оскільки вираз 1=1 завжди істинний).

Мова SQL дозволяє об'єднувати результати декількох запитів за допомогою оператора UNION. Це надає зловмисникові можливість отримати несанкціонований доступ до даних.

Розглянемо скрипт відображення новини (ідентифікатор новини, яку необхідно відобразити, передається в параметрі id):

```
$res = mysql_query("SELECT id_news, header, body, author FROM news WHERE id_news = ". $_REQUEST['id']).
```

Якщо зловмисник передасть як параметр id конструкцію -1 UNION SELECT 1,username, password,1 FROM admin, це викличе виконання SQL-запиту:

```
SELECT id_news, header, body, author FROM news WHERE id_news =-1 UNION SELECT 1,username,password,1 FROM admin.
```

Оскільки новини з ідентифікатором -1 завідомо не існує, з таблиці news не буде вибрано жодного запису, проте в результат потраплять записи, несанкціоновано відібрані з таблиці admin внаслідок ін'єкції SQL.

Розщеплення SQL-запиту.

Для розділення команд в мові SQL використовується символ ; (крапка з комою), включаючи цей символ до запиту, зловмисник отримує можливість виконати декілька команд в одному запиті. Наприклад, якщо в параметри скрипта

```
$id = $_REQUEST['id'];
$res = mysql_query("SELECT * FROM news WHERE id_news = $id");
```

зловмисником передається конструкція, що містить крапку з комою, наприклад, 12;INSERT INTO admin (username, password) VALUES ('HaCkEr', 'foo'); то в одному запиті будуть виконані 2 команди

```
SELECT * FROM news WHERE id_news = 12;
```

INSERT INTO admin (username, password) VALUES ('HaCkEr', 'foo');

і в таблицю admin буде несанкціоновано доданий запис HaCkEr.

IV. Атаки типу впровадження SQL-коду

Пошук скриптів, уразливих для атаки.

На цьому етапі зловмисник вивчає поведінку скриптів сервера при маніпуляції вхідними параметрами з метою виявлення їх аномальної поведінки. Маніпуляція відбувається всіма можливими параметрами [2,3]:

– даними, переданими через методи POST і GET;
– значеннями [HTTP-Cookie];
– HTTP_REFERER (для скриптів);
– AUTH_USER та AUTH_PASSWORD (при використанні автентифікації).

Як правило, маніпуляція зводиться до підстановки в параметри символу одинарної (рідше подвійної або зворотної) лапки.

Аномальною поведінкою вважається будь-яка поведінка, при якій сторінки, що одержані до і після підстановки лапок, розрізняються (і при цьому немає повідомлення «неправильний формат параметрів»).

Найчастіші приклади аномальної поведінки:

– виводиться повідомлення про різні помилки;
– при запиті даних, запитувані дані не виводяться взагалі, хоча сторінка відображається і т. д.

Слід враховувати, що відомі випадки, коли повідомлення про помилки, в силу специфіки розмітки сторінки, не видно в браузері, хоча і присутні в її HTML-коді.

V. Захист від атак типу впровадження SQL-коду

Фільтрація рядкових параметрів [2,3].

Припустимо, що код, який генерує запит (на мові програмування Паскаль), виглядає так:

```
statement:= 'SELECT * FROM users WHERE name = ''' + userName + ''';'.
```

Щоб впровадження коду було неможливо, для деяких систем управління базами даних, в тому числі, для MySQL, потрібно брати в лапки всі рядкові параметри. У самому параметрі замінюють лапки на \", апостроф на \', зворотну косу риску на \\ (це називається «екранувати спецсимволи»). Це можна робити, використовуючи такий код:

```
statement:= 'SELECT * FROM users WHERE name = ' + QuoteParam(userName) + ';';
```

для PHP фільтрація може бути такою:

```
<?php
$query = "SELECT * FROM users WHERE user='".mysql_real_escape_string($user)."';";
?>
```

Фільтрація цілочисельних параметрів [2,3].

Візьмемо інший запит:

```
statement:= 'SELECT * FROM users WHERE id = ' + id + ';'.
```

У цьому випадку поле id має числовий тип, і його найчастіше не беруть в лапки. У такому випадку допомагає перевірка – якщо змінна id не є числом, запит взагалі не повинен виконуватися.

Наприклад, на Delphi для протидії таким ін'єкціям використовується код:

```
id_int:= StrToInt(id);
statement:= 'SELECT * FROM users WHERE id = ' + IntToStr(id_int) + ';'.
```

У випадку помилки функція StrToInt викличе виняток класу EConvertError, і в його обробнику можна буде вивести повідомлення про помилку. Подвійне перетворення забезпечує коректну реакцію на числа у форматі $132AB (шістнадцяткова система числення). На стандартному Паскалі, який не вміє обробляти виняток, код дещо складніший.

Для PHP цей метод буде виглядати так:

```
$query = 'SELECT * FROM users WHERE id = '. intval($id).
```

Усікання вхідних параметрів [2,3].

Для внесення змін в логіку виконання SQL-запиту потрібно впровадження достатньо довгих рядків. Так, мінімальна довжина такого рядка у наведених вище прикладах становить 8 символів («1 OR 1=1»). Якщо максимальна довжина коректного значення параметра невелика, то одним з методів захисту може бути максимальне усікання значень вхідних параметрів.

Наприклад, якщо відомо, що поле id у вищенаведених прикладах може приймати значення не більше 9999, можна «відрізати зайві» символи, залишивши не більше чотирьох:

```
statement:= 'SELECT * FROM users WHERE id = ' + LeftStr(id, 4) + ';'.
```

Висновок

Актуальність проблеми безпеки веб-додатків зростає з кожним днем. Більшість вразливостей, що існують на даний момент в цій сфері, пов'язані з помилками і недоліками, допущеними на етапі розробки сайту. В цій роботі були досліджені атаки типу SQL-ін’єкція. Наведені приклади реалізації, методи впровадження ін’єкції та основні методи захисту від впровадження SQL-коду. Метою подальших робіт є розробка атак типу SQL-ін’єкція на спеціалізовані веб-додатки та розробка методів захисту від них на мові програмування Golang.

Література

[1] Dafydd Stuttard, Marcus Pinto. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws 2 edition Wiley. 2011. 912c.

[2] Скембрейц Дж. Безопасность Web-приложсний – готовые решения / Дж. Сксмбрейц, М. Шема. – М.: Издательский дом «Вильямс», 2003. – 384 с.

[3] Habrahabr [Електронний ресурс] Методы обхода защитных средств веб-приложений при эксплуатации SQL-инъекций https://habr.com/ru/post/326362/ [дата звернення 06.10.2019].

Аналіз функціональності платіжного протоколу Lightning Network

Анастасія Сапожкова[1], Олександр Сєвєрінов[1], Андрій Власов[2]

1. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м. Харків, пр. Науки, 14,
E-mail: anastasiia.sapozhkova@nure.ua,
E-mail: oleksandr.sievierinov@nure.ua

2. Науковий центр Повітряних Сил, Харківський національний університет Повітряних Сил, УКРАЇНА, м. Харків, вул. Клочківська, 228,
E-mail: vav_and@i.ua

***Коротка анотація** – Lightning is a decentralized network using smart contract functionality in the blockchain to enable instant payments across a network of participants. The Lightning Network is dependent upon the underlying technology of the blockchain. By using real blockchain transactions and using its native smart-contract scripting language, it is possible to create a secure network of participants which are able to transact at high volume and high speed.*

Ключові слова – Lightning Network; платіжна мережа; смарт-контракт; блокчейн.

I. Вступ

Lightning Network - це технологія, що дозволяє трансформувати будь-яку криптовалюту в швидкий і надійний платіжний засіб з мінімальними комісіями. Її розробили двоє програмістів: Джозеф Пун і Тедж Дрійа в 2016 році, активна фаза тестування і запуску почалася тільки в кінці 2017 року. Lightning Network планується використовувати для проведення мікроплатежів у біткоїні (і інших криптовалютах) і розвантаження його основної мережі. Завдяки використанню цієї технології кількість транзакцій в секунду може збільшитися в сотні тисяч разів, а комісія за переказ зведеться до мінімальних значень або зовсім буде дорівнювати нулю.

Типова транзакція в мережі біткоін виглядає так: відправник хоче перевести певну суму в BTC одержувачу і щоб все пройшло гладко, він повинен заплатити чималу комісію, яка йде іншому учаснику мережі в якості нагороди за включення транзакції в блок. Транзакція стає в чергу - відправник і одержувач чекають, поки вона запишеться в блок, і тільки тоді отримують кінцевий результат. Все це є побічними ефектами малого розміру блоку (всього 1 Мб для біткоін-мережі) і величезної кількості транзакцій в черзі [1].

Lightning Network вирішує цю проблему за допомогою так званих «каналів оплати». Технологія дозволяє налаштувати окремі мережі між двома учасниками основної мережі, внутрішні транзакції яких не будуть записуватися в блокчейн, а тільки зберігатися в рамках мережі каналу. Це означає, що користувачі можуть здійснювати скільки завгодно транзакцій між собою, не завантажуючи основну мережу.

При роботі за такою схемою в блокчейн записується тільки факт відкриття і закриття каналу с вказаним початковим і кінцевим балансом сторін, а всі транзакції, що пройшли по каналу, не займають місце у блокчейні. Така мережа може працювати не тільки з двома учасниками - вона запросто масштабується до будь-якого розміру, поєднуючи відправників і одержувачів ланцюжком в один канал, вибираючи оптимальний маршрут. Важливий момент: мережа пересилає по каналам не кошти, а тільки інформацію про володіння ними, тобто, згадану вище розписку, адже для відкриття каналу потрібне внесення сторонами певного депозиту.

II. Вирішення проблеми та результати

Network - це peer-to-peer платіжна мережа для проведення мікротранзакцій, що підтримує такі криптовалюта, як Bitcoin, Ethereum, Litecoin. Завданням цієї мережі є прискорення кріптовалютних платежів без делегування володіння грошима третій стороні, а також об'єднання різних криптовалюта в єдину мережу з прикордонними точками у вигляді децентралізованих бірж.

Двома елементами будь-peer-to-peer мережі є вузол і з'єднання:

– під вузлом в Lightning Network може розумітися мобільний додаток / десктоп програма / серверне ПЗ, яке підтримує протокол спілкування Lightning Network. Одним з таких прикладів є реалізація на мові Go;

– під з'єднанням в Lighting Network розуміється платіжні канал - відношення між учасниками, яке реєструється в блокчейн і регулюється смарт-контрактом.

Кожен вузол має можливість приймати і відправляти платежі, а також виступати в ролі провідника платежів, отримуючи за це комісію. Надіслати платіж від одного учасника мережі до іншого можна тільки в разі наявності шляху, що складається з платіжних каналів, що з'єднує одержувача і відправника. За допомогою можливості взяття комісії за проведення платежу, в мережі стимулюється створення вузлів, які з'єднують між собою безліч інших користувачів. В якості таких вузлів в майбутньому можуть виступати біржі і онлайн гаманці, а також інші організації, які будуть будувати свої послуги на основі даної технології.

Відмінність Lightning Network від таких мереж як Visa і MasterCard полягає в тому, що приєднатися до неї може будь-хто. Досягається це властивість "відкритості" за допомогою використання смарт-контрактів. Також варто врахувати, що в Lightning Network вбудовані алгоритми, подібні мережі Tor, тому одержувач і відправник платежу не відомі вузлів-провідникам. Як саме досягаються ці властивості, описано в технічних частинах даного циклу статей.

Lightning Network може бути застосована до будь-якого блокчейну, який підтримує деякі основні можливості, такі як мультипідписні транзакції, мітки часу та основні смарт контракти.

Якщо LN розташована на вершині мережі біткойн, мережа біткойнів може значно збільшити ємність, конфіденційність, деталізацію та швидкість, не жертвуючи принципами довірчої операції без посередників.

Конфіденційність. Платежі мережі Lightning Network є набагато приватнішими, ніж платежі на блокчейні біткойн, оскільки вони не є загальнодоступними. Хоча учасники маршруту можуть бачити платежі, поширювані по їх каналах, вони не знають відправника або одержувача.

Перемінність. LN робить набагато складніше застосовувати спостереження та чорні списки на біткойні, що підвищує взаємозв'язок валюти.

Швидкість. Транзакції Bitcoin з використанням мережі Lightning Network вирішуються за мілісекунди, а не за хвилини, оскільки HTLC видаляються без здійснення транзакцій до блоку.

Гранулярність. LN може дозволити платежі, принаймні настільки малі, як обмеження "пилу" біткойн, можливо, навіть менше. Деякі пропозиції дозволяють збільшувати субстатоші.

Потужність. LN збільшує місткість біткойн-системи на кілька порядків. Немає практичної верхньої межі кількості платежів в секунду, яку можна прокладати через LN, оскільки це залежить тільки від потужності та швидкості кожного вузла.

Безтурботна операція. LN використовує транзакції bitcoin між вузлами, які працюють як рівні, не довіряючи один одному. Таким чином, мережа Lightning зберігає принципи роботи системи біткойн, значно розширюючи її робочі параметри.

Як згадувалося раніше, протокол мережі Lightning Network - це не єдиний спосіб запровадження маршрутизованих платіжних каналів. Інші пропоновані системи включають Tumblebit і Teechan. Однак у цей час LN вже була розгорнута на тест-мережі. Кілька різних команд розробили конкуруючі реалії LN і працюють над єдиним стандартом сумісності (так званий BOLT). Цілком імовірно, що LN буде першою мережею каналів розрахункових каналів, яка буде розгорнута у виробництві.

У 2015 році проект Bitcoin зіткнувся з проблемою, яка полягала в масштабуванні. Мережа не могла обробляти більшу кількість транзакцій щосекунди, ніж встановлений ліміт в 3-7 операцій. Цей ліміт закладений в саму концепцію створення платформи, а саме збільшення показників пропускної здатності системи за допомогою збільшення блоків призвело б до зниження ефективності принципів децентралізованого управління криптовалютою, оскільки єдина точка відмови відсутня.

Розробники застосували для вирішення даної проблеми кілька різних схем, що в результаті призвело до створення форку Bitcoin Cash і поділу блокчейна, при цьому сам Bitcoin почав використовувати масштабування через Lightning Network.

Таким чином, за допомогою опису функціональності продукту Lightning Network, йому можна дати наступне визначення: проект являє собою розробку, орієнтовану на підвищення пропускних показників біткоін-мережі при досягненні частоти операцій, яку можна порівняти з показниками мережі Visa.

Висновок

Останні кілька років технологія блокчейн все більше застосовується сучасним суспільством. Блокчейн – розподілений електронний реєстр, в якій реєструється кожна транзакція, яка відбувається в системі. Блокчейн має змогу вирішувати проблеми такі як: безпека, висока доступність та швидкість виконання транзакцій[2].

Як було вже зазначено, досить часто користувачами даної технології є цифрові валюти, не виняток і найвідоміша з них – біткоін. Ринок цифрових валют досить швидко розвивається, так максимальна капіталізація ринку криптовалют складала 190 млрд. доларів. Проте це не єдине ефективне застосування цієї технології, так, ринок стартапів на базі використання технології blockchain, за оцінками експертів, залучить у 2018 році інвестицій на суму 3 млрд. доларів, що цілком впевнено робить технологію альтернативою традиційним венчурним інвестиціям.

Варто звернути увагу, що використання технології блокчейн викликає зацікавлення і в Україні. Так стало відомо, що Україна уклала угоду з міжнародною технологічною компанією Bitfury Group про переведення всіх електронних державних даних на блокчейн.

Таким чином стає зрозуміло, що технологія Blockchain є певною мірою революційною та може бути використана в різних сферах людської діяльності. Зокрема, мова йде про такі реалізації як: криптовалюти, різного роду реєстри, корпоративного, або ж державного значення. Але у криптовалют, зокрема у біткойна, є великий недостаток – пропускна здатність. Через це виникає велика кількість технологій, що намагаються виправити цю проблему. Найвиразнішою та найперспективнішою з таких є технологія Lightning Network.

Література

[1] Andreas M. Antonopoulos Mastering Bitcoin: Unlocking Digital Cryptocurrencies / Andreas M. Antonopoulos – K. : NGITS, 2014. – С. 150 – 290.

[2] Don Tapscott, Alex Tapscott Blockchain Revolution: How the Technology Behind Bitcoin is Changing Money, Business, and the World / Don Tapscott, Alex Tapscott Blockchain – K. : Information Systems, 2016 – С. 100 – 150.

Захист DHCP сервера в мережі підприємства

Сергій Сапанович[1], Віталій Мартовицький[2]

1. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, г. Харків, пр. Науки, 14, E-mail: serhii.sapanovych@nure.ua

2. Кафедра електронних обчислювальних машин, Харківський національний університет радіоелектроніки, УКРАЇНА, г. Харків, пр. Науки, 14, E-mail: vitalii.martovytskyi@nure.ua

This paper addresses the basic network security issues at the OSI link layer network. Different types of attacks on local computer networks are analyzed, as well as modern methods of their detection and prevention. The most promising areas for intrusion detection systems based on the use of Cisco Catalyst Switches are offered.

Ключові слова – інформаційні системи, мережа, Cisco Systems, DHCP, security, ARP.

I. Вступ

У сучасних локальних мережах обмін інформацією, як правило, передбачає передачу даних через комутатор. Тому сам комутатор і протоколи, які використовують комутатори можуть бути метою атак. Більш того, деякі настройки комутаторів (як правило, це налаштування за замовчуванням) дозволяють виконати ряд атак і отримати несанкціонований доступ до мережі або вивести з ладу мережеві пристрої. Однак, комутатор може бути і досить потужним засобом захисту. Так як через нього відбувається все взаємодії в мережі, то логічно контролювати це на ньому. Звичайно, використання комутатора як засобу захисту передбачає, що використовується не найпростіший комутатор 2-го рівня, а комутатор з відповідними функціями для забезпечення безпеки. В даній доповіді проведемо аналіз безпеки комутатора Cisco Catalyst 2960, використовуючи додаткові вбудовані функції, для забезпечення необхідних політик безпеки мережі.

II. Основний матеріал

Розглянемо докладніше функції комутаторів Cisco Catalyst для забезпечення безпеки мережі. Список їх наведено в табл. 1.

IP Source Guard (Dynamic IP Lockdown) – функція комутатора, яка обмежує IP-трафік на інтерфейсах 2-го рівня, фільтруючи трафік на підставі таблиці прив'язок DHCP snooping і статичних відповідностей. Функція використовується для боротьби з IP-spoofingом.

Port security – функція комутатора, що дозволяє вказати MAC-адреси хостів, яким дозволено передавати дані через порт. Після цього порт не передає пакети, якщо MAC-адресу відправника не вказано як дозволену. Крім того, можна вказувати не конкретні MAC-адреси, дозволені на порту комутатора, а обмежити кількість MAC-адрес, яким дозволено передавати трафік через порт. Використовується для запобігання [1]:

− несанкціонованої зміни MAC-адреси мережевого пристрою або підключення до мережі;

– атак спрямованих на переповнення таблиці комутації.

DHCP snooping – функція комутатора, призначена для захисту від атак з використанням протоколу DHCP. Наприклад, атаки з підміною DHCP- сервера в мережі або атаки DHCP starvation, яка змушує DHCP-сервер видати всі існуючі на сервері адреси зловмисникові. DHCP snooping регулює тільки повідомлення DHCP і не може вплинути безпосередньо на трафік користувачів або інші протоколи. Деякі функції комутаторів, що не мають безпосереднього відношення до DHCP, можуть виконувати перевірки на підставі таблиці прив'язок DHCP snooping (DHCP snooping binding database).

Для правильної роботи DHCP snooping, необхідно вказати які порти комутатора будуть довіреними (trusted), а які - ні (untrusted, в подальшому - ненадійними) [2]:

Ненадійні (Untrusted) – порти, до яких підключені клієнти. DHCP- відповіді, що приходять з цих портів відкидаються комутатором.

ТАБЛИЦЯ 1

Функції комутаторів для забезпечення безпеки роботи мережі на канальному рівні

Функція комутатора	Від яких атак захищає
Port security	Переповнення таблиці комутації, несанкціонована зміна MAC-адреси
DHCP Snooping	Підміна DHCP-сервера в мережі, DHCP starvation
Dynamic ARP Inspection	ARP-spoofing
IP Source Guard	IP-spoofing

Для ненадійних портів виконується ряд перевірок повідомлень DHCP і створюється база даних прив'язки DHCP (DHCP snooping binding database).

Довірені (Trusted) – порти комутатора, до яких підключений інший комутатор або DHCP-сервер. DHCP-пакети отримані з довірених портів, не відкидаються.

За замовчуванням комутатор відкидає DHCP-пакет, який прийшов на ненадійний порт, якщо:

– приходить одне з повідомлень, які відправляє DHCP-сервер (DHCPOFFER, DHCPACK, DHCPNAK або DHCPLEASEQUERY);

– приходить повідомлення DHCPRELEASE або DHCPDECLINE, в якому міститься MAC-адресу з бази даних прив'язки DHCP, але інформація про інтерфейс в таблиці не збігається з інтерфейсом, на якому був отриманий пакет;

– у DHCP-пакеті, що прийшов, не збігаються MAC-адреса, вказана в DHCP-запиті, і MAC-адреса відправника;

– приходить DHCP-пакет, в якому є опція 82.

Dynamic ARP Inspection – функція комутатора, призначена для захисту від атак з використанням протоколу ARP [3]. Наприклад, атаки ARP-spoofing, що дозволяє перехоплювати трафік між вузлами, які розташовані в межах одного широкомовного домену. Dynamic ARP Inspection регулює тільки повідомлення протоколу ARP і не може вплинути безпосередньо на трафік користувачів або інші протоколи.

Для правильної роботи Dynamic ARP Inspection, необхідно вказати які порти комутатора будуть довіреними (trusted), а які – ні (untrusted) [10]:

– ненадійні (Untrusted) – порти, до яких підключені клієнти. Для ненадійних портів виконується ряд перевірок повідомлень ARP.

– довірені (Trusted) – порти комутатора, до яких підключений інший комутатор. Повідомлення протоколу ARP отримані з довірених портів, не відкидаються.

Якщо порт ненадійний, комутатор перехоплює все ARP-запити і ARP- відповіді на ненадійних портах перш ніж перенаправляти їх. Комутатор перевіряє відповідність MAC-адреси IP-адресі на ненадійних портах. Перевірка відповідності MAC-адреси IP-адресі може виконуватися на підставі статичних записів або бази даних прив'язки DHCP.

Висновки

У роботі розглянуті основні проблеми, пов'язані з безпекою мереж на канальному рівні моделі OSI. Проаналізовано різні типи атак на локальні комп'ютерні мережі, а також сучасні методи їх виявлення і попередження. Запропоновано найбільш перспективні напрямки розвитку систем виявлення вторгнень, засновані на використанні вбудованих функцій безпеки комутаторів Cisco Catalyst.

Також показано, що без використання вбудованих функцій безпеки комутаторів всі атаки, при невеликих витратах часу, дозволили домогтися бажаного ефекту (відмова в обслуговуванні/перехоплення інформації). Після детальної настройки захисних функцій атаки виявилися абсолютно неефективними. Окремі сценарії мережевих атак, а також механізми їх виявлення були реалізовані програмно за допомогою засобів мережевого емулятора Cisco Packet Tracer 5.3.2.

Література

[5] Брюс Александер, Тони Аллен, Матт Карлинг и др. Руководство по технологиям объединенных сетей Cisco. Изд. 4-е. – М.: Издательский дом «Вильямс», 2005. – 1040 с.: ил.

[6] Официальный сайт Cisco Systems. Программа Cisco Packet Tracer [Електроний ресурс]. Режим доступу: http://www.cisco.com/web/learning/netacad/course_catalog/PacketTracer.html.

[7] Джеймс Бони. Руководство по Cisco IOS. – СПб.: Питер, М: Издательство «Русская редакция», 2008. – 784 с.: ил.

Безпека технології блокчейн для децентралізованих систем

Данило Скічко[1], Тетяна Гріненко[2], Олексій Нарєжній[3]

1. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м. Харків, пр. Науки, 14, E-mail: meksvinz@gmail.com

2. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м. Харків, пр. Науки, 14, E-mail: tetiana.grinenko@nure.ua

3. Кафедра безпеки інформаційних систем і технологій, Харківський національний університет імені В.Н. *Каразіна*, УКРАЇНА, м. Харків, майдан Свободи, 4, E-mail: o.nariezhnii@karazin.ua

The analysis of blockchain technology security and analysis of existing systems to identify existing vulnerabilities and possible attacks on decentralized systems are given. The work provides examples of technologies built on the blockchain and evaluates their software features.

Атака, безпека, блокчейн, вразливість, криптовалюта, технологія, bitcoin, monero.

I. Вступ

Сьогодні дуже популярною є помилкова думка, що технології, які побудовані на блокчейні майже не мають вразливостей, а клієнти, які користуються такими технологіями, не можуть понести матеріальних або будь-яких інших видів збитків. Але це не так. Блокчейн сам по собі – це всього лише підхід до зберігання даних, а безпека технологій, побудованих на ньому, в більшості випадків залежить тільки від програмної реалізації останніх.

Блокчейн має на увазі децентралізацію, а децентралізація, в свою чергу, безліч вузлів і клієнтів в мережі, які постійно взаємодіють один з одним [1]. Найчастіше, до такої мережі може приєднається кожен охочий (запустивши попередньо на своєму вузлі відповідне програмне забезпечення).

До технології, що побудована на блокчейні, висуваються високі вимоги безпеки. Такі технології повинні мати якомога менше програмних вразливостей, що можуть бути використані зловмисником, яким потенційно може бути будь-який новий член мережі. Варто зауважити, що проблема високих вимог до мережі частково вирішується формуванням довіреного кола вузлів, які будуть підтримувати блокчейн, тим самим не даючи зловмисникам використовувати уразливості навіть за їх наявності.

II. Вразливості блокчейн

Переважна більшість технологій, що побудовані на блокчейні – це криптовалюти [1]. Одна з найвідоміших вразливостей криптовалют – це атака 51%. Суть її полягає в тому, що зловмисники отримують контроль над мережею, шляхом володіння обчислювальною потужністю, яка складає 51% від потужності всієї мережі.

При успішному використанні цієї вразливості, зловмисники можуть фактично диктувати дані, які будуть додані в блокчейн в майбутньому. За допомогою цієї уразливості, в контексті криптовалют, можна провести так звану атаку подвійної витрати. Варто зауважити, що на сьогодні досить багато проектів закрилися на самому старті саме через те, що зловмисники досить легко використовували цю вразливість (на старті проекту у мережі мало вузлів і сумарна потужність не надто велика).

У блокчейн рішеннях користувач є відповідальним за доступність своїх даних, а атрибутом доступу до системи зазвичай виступає ключова пара. Ще одна вразливість – крадіжка або втрата ключової пари. Фактично факт володіння активами в блокчейні підтверджується особистим ключем користувача. При крадіжці ключа зловмисник зможе безперешкодно використовувати активи користувача. Ще одним недоліком є те, що користувач, маючи на руках всі докази втрати або крадіжки ключів, ніяк не зможе їх відновити (так буває при втраті або компрометації пароля якогось сервісу). Тому користувачі децентралізованих систем, побудованих на блокчейні, повинні розуміти відповідальність за збереження своїх даних і враховувати не тільки особливості безпеки технології, а й уразливості своїх локальних машин або сховищ даних.

Недоліки, пов'язані з втратою особистих ключів, вирішуються за допомогою використання сторонніх сервісів, які виступають в ролі посередника між користувачем і кінцевою системою. Тим самим вони беруть на себе відповідальність за збереження особистих даних користувачів. З одного боку, це полегшує життя кінцевим користувачам, а з іншого, подібні сервіси вже не раз піддавалися атакам з наступною крадіжкою ключових даних (що відповідно є великою шкодою для клієнтів). Тому для здійснення транзакції в системі оптимальним рішенням буде використання мультипідпису або смартконтрактів.

Крім вразливостей, пов'язаних з аспектом децентралізації та ключових пар, у криптовалют є програмні уразливості або особливості, які, наприклад, дозволяють сторонньому спостерігачеві провести аналіз дій користувача і дізнатися конфіденційну інформацію про нього. Наприклад, в Bitcoin всі транзакції і адреси відкриті всім користувачам мережі і стороннім спостерігачам. При належних зусиллях можна зіставити адреси, з яких проводилися транзакції і події в реальному світі (наприклад, покупка автомобіля), та спробувати дізнатися особистість клієнта системи або інформацію, яка в кінцевому підсумку призведе до клієнта. Крім простого аналізу до Bitcoin застосовні й інші алгоритми, які дозволяють деанонімізувати користувача.

III. Децентралізована облікова система Monero

Однак не всі криптовалюти працюють так як Bitcoin. Адже криптовалюта – це всього лише програмна реалізація протоколу, який працює на блокчейні. В одних протоколах приділяється увага відкритості та прозорості операцій, тоді як в інших, приділяється увага безпеці користувачів системи (що найчастіше ускладнює їм життя на благо збереження їх особистих даних). Прикладом такої криптовалюти є проект Monero. Децентралізована облікова система Monero – це система, в якій упор зроблений в першу чергу на забезпечення анонімності користувача [2].

В системі Monero така інформація як: суми транзакцій, ідентифікаційні дані відправника та ідентифікаційні дані одержувача приховані в ланцюжку блоків, тому дії по зберіганню і витраті монет не можна відстежити. Monero також включає в себе модуль wallet (гаманець). Гаманець зберігає ключі і здійснює складні криптографічні операції з управління активами. Для доступу до гаманця необов'язково контролювати приватні ключі, досить зберегти seed фразу, яка була використана при генерації гаманця. Seed – це секретне число, яке гаманець використовує, щоб згенерувати ключі та отримати доступ до монет, хоча для зручності і сприйняття людиною це число перетворюється в серію з 12-25 слів [2]. Seed фраза (або основний секрет) повинен бути доступний тільки користувачеві гаманця, так як знання seed-фрази гаманця визначає володіння активами, які гаманець зберігає.

В Monero забезпечуються розширені функціональні можливості, анонімність і конфіденційність завдяки використанню декількох унікальних криптографічних технологій, які захищають користувачів і їх діяльність від публічного доступу, таких як:

– RingCT (ring confidential transactions) – приховує суму транзакції;

– Ring signatures – дозволяє захистити користувача від розкриття виходу, який був витрачений;

– Stealth address – гарантує, що адреса отримувача не буде записана в ланцюжку блоків;

– Kovri – це реалізація I2P, написана на C++, яка дозволяє розірвати зв'язок між транзакціями і фізичним місцем розташування, приховуючи мережеві ознаки активності вузла Monero.

RingCT – це криптографічна технологія, яка приховує кількість грошей, що надійшли в будь-якій транзакції. У більшості криптовалют суми транзакцій відправляються у вигляді відкритого тексту, який є видимим будь-якому спостерігачеві. RingCT зберігає цю конфіденційну інформацію в секреті, дозволяючи відправнику довести, що у нього достатньо грошей для транзакції, не розкриваючи значення цієї суми. Це можливо завдяки криптографічним зобов'язаннями і range proofs. Range proofs – ще один важливий механізм в RingCT, як метод, який гарантує, що відправляєма кількість монет більше нуля і менше певного числа. Це необхідно для запобігання відправки користувачем від'ємних або неймовірно великих сум.

Ring signatures (кільцеві підписи) – це функція Monero, яка використовується для захисту відправника транзакції шляхом маскування джерела витрачених монет (не витраченого виходу). Маскування забезпечується шляхом перемішування ключів кількох виходів в ланцюжку блоків. У результаті можна перевірити, що один з виходів був витрачений, але який саме визначити практично неможливо. Кільцеві підписи за замовчуванням застосовуються до кожної транзакції.

Всі транзакції в Monero використовують stealth addresses для захисту конфіденційності одержувача, тобто для запобігання запису адреси одержувача в ланцюжок блоків. Кожна транзакція в Monero відправляється на унікальну одноразову адресу, доступ до якої має тільки той, кому призначалася транзакція.

Kovri – це функція протоколу Monero, яка призначена для захисту від розкриття IP адреси вузла. Трафік, що виходить від вузла мережі Monero, проходить через інтернет-провайдера, який може виділити активність вузла і ідентифікувати його як вузол мережі Monero.

Висновок

Відкритість транзакцій в децентралізованих системах, з одного боку, є плюсом для спільноти, а з іншого – з'являються ризики розкриття адреси суб'єкта. Аналіз існуючих децентралізованих систем (наприклад, Bitcoin, Ethereum) дозволяє зробити висновок, що на основі ланцюжка блоків можна ідентифікувати приналежність деяких адрес конкретним суб'єктам. Тому для здійснення транзакції в системі оптимальним є використання мультипідпису або смартконтрактів.

Література

[1] Кравченко П. Блокчейн и децентрализованные системы: учебное пособие для студ. заведений высш. Образования : в 3 частях. Ч. 1 / П. Кравченко, Б. Скрябин, О. Дубинина. – Харьков : ПРОМАРТ, 2018. – 400 с. – ISBN 978-617-7634-26-2.

[2] Mastering Monero [Электронный ресурс] – https://github.com/monerobook/monerobook

Виявлення аномалій в SQL запитах

Владислав Степанов[1], Віталій Мартовицький[2]

1. Кафедра безпеки інформаційних технологій,
Харківський національний університет радіоелектроніки,
УКРАЇНА, г. Харків, пр. Науки, 14,
E-mail: vladyslav.stepanov@nure.ua

2. Кафедра електронних обчислювальних машин,
Харківський національний університет радіоелектроніки,
УКРАЇНА, г. Харків, пр. Науки, 14,
E-mail: vitalii.martovytskyi@nure.ua

***Коротка анотація** – The paper deals with the possibility of using machine learning for tasks of classification of attacks not web system. A method for detecting anomalies in the form of SQL injection with the use of an autoencoder with Seq2Seq architecture has been developed, and the testing of the operation of the given transactions is performed.*

Ключові слова – SQL-ін'єкція, WEB-ресурси, SQL, машинне навчання, автоенкодер.

I. Вступ

Сучасне підприємство має розвинену мережеву інфраструктуру, в якій працюють корпоративні інформаційні системи, що забезпечують підтримку всіх бізнес-процесів організації. Головним джерелом бізнес-інформації в таких мережевих інфраструктурах є сховища та бази даних, в яких зберігається внутрішня оперативна та фінансова інформація, персональні дані співробітників, інформація про замовників та клієнтів, інтелектуальна власність, дослідження ринку та аналіз діяльності конкурентів, платіжна інформація.

За деякими даними, в промислово розвинених країнах середній збиток від одного злочину в сфері комп`ютерної інформації становить приблизно 450 тис. дол., а щорічні сумарні втрати в США і Західній Європі, за даними, що наводять В. Гайкович та А. Прешин, досягають 100 млрд. і 35 млрд. дол. В останні десятиріччя зберігалася стійка тенденція до зростання збитків, пов'язаних із злочинністю в сфері комп'ютерної інформації. В пресі та літературі наведено багато подібних прикладів [1].

Сьогодні більшість фірм виробників систем управління сховищами та базами даних намагаються удосконалити засоби захисту, але їхні зусилля, як правило, скеровані тільки на усунення відомих вразливостей власних продуктів.

Основними напрямками розвитку технологій систем управління базами даних (СКБД) є реляційні та нереляційні СКБД.

Враховуючи все вищезазначене, актуальною задачею є комплексне дослідження і систематизація питань захисту сховищ та баз даних з урахуванням загальних тенденцій розвитку підходів до забезпечення інформаційної безпеки та усунення загроз з використанням технології інтелектуального аналізу даних.

Це обумовлено, наприклад, легкістю виявлення і експлуатування уразливостей баз даних і, як наслідок, їх використання зловмисниками, що івдображено в статтях [2-4]. Тому, побудова моделей і методів для виявлення SQL-ін'єкцій у веб-ресурсах є актуальним завданням.

II. ОСНОВНИЙ МАТЕРІАЛ

Проблема виявлення веб-атак розглядається з точки зору виявлення аномалій. На етапі навчання моделі видаються тільки нормальні HTTP-запити. На етапі тестування модель визначає, чи отриманий запит аномальним чи ні.

Для виявлення аномалій в HTTP-запиті використовується архітектура Seq2Seq. Модель Seq2Seq [5] складається з двох багатошарових LSTM - кодера і декодера. Кодер відображає вхідну послідовність в вектор фіксованої довжини. Декодер декодує цільової вектор, використовуючи вихід кодера. При навчанні автоенкодер є модель, в якій цільові значення встановлюються такими ж, як вхідні значення.

Ідея полягає в тому, щоб навчити мережу декодувати речі, які вона бачила, або, іншими словами, наближати тотожне відображення. Якщо навченому автоенкодеру дають аномальний зразок, він, ймовірно, відтворює його з високим ступенем помилки, просто тому, що ніколи його не бачив. Структура автоенкодеру представлена на рисунку 1.

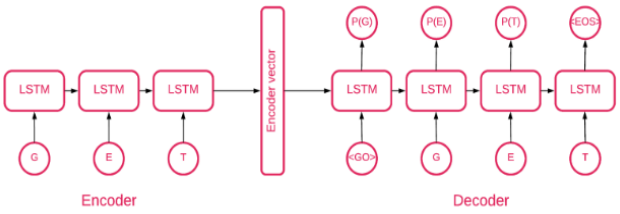


Рисунок 1 – Архітектура Seq2Seq

У відповідності до рисунку дані вводу кодера представлені буквами G, E, T, а дані вводу декодера – GO, P(G), P(E), P(T), <EOS>.

Представлене рішення складається з декількох частин: ініціалізація моделі, навчання, прогнозування та перевірка.

Модель створюється як екземпляр класу Seq2Seq, який має такі аргументи конструктора:

– batch_size - число вибірок в пакеті

– embed_size - розмір простору вбудовування (повинен бути меншим за розмір словника)

– hidden_size - кількість прихованих станів у lstm

– num_layers - кількість блоків lstm

– контрольні точки - шлях до каталогу контрольно-пропускних пунктів

– std_factor - кількість stds, яка використовується для визначення порогу моделі випадання - ймовірність збереження кожного елемента

– vocab - об'єкт лексики

Далі ініціалізуються шари автоенкодера. Спочатку кодер потім декодер

Так як проблема, яку вирішуємо, полягає у виявленні аномалій, цільові значення і вхідні дані збігаються. Далі на основі навчальної вибірки відбувалося навчання нашого автоенкодеру. Навчальна вибірка містить дані з 21991 нормальними і 1097 аномальними HTTP-запитами з банківських додатків, яку було отримано з ресурсу Kagel.

Після кожної епохи найкраща модель зберігається в якості контрольної точки, яку потім можна завантажити. З метою тестування було створено веб-додаток, який було захищено розробленою моделлю, щоб перевірити, чи будуть реальні атаки успішними. На етапі тестування на нашій відкладеної вибірці ми отримали дуже хороші результати: precision і recall близькі до 0,99. та ROC-крива наближається до 1, що показано на рис. 2

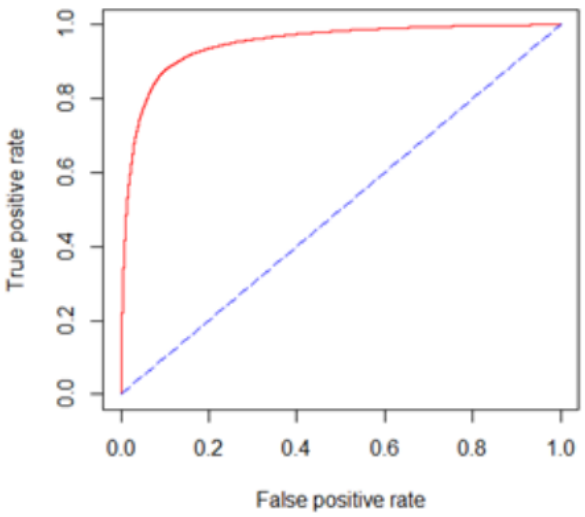


Рисунок 2 – Графік ROC-кривої

Висновки

У роботі розглядається можливість застосування машинного навчання для задач класифікації атак не веб системи.

Розроблено метод виявлення аномалій у вигляді SQL-ін'єкції з використанням автоенкодеру з архітектурою Seq2Seq та протестовоно його роботу на даних транзакцій банківської програми.

Запропонована система оцінки з набором даних, що мають різні запити. Отриманий результат показує, що запропонована система здатна виявити шкідливий запит і, отже, класифікувати користувачів. Надалі цей тип системи також може використовуватися для класифікації атак ботнетів та різних систем безпеки шляхом порівняння відмінностей між запитами, поданими звичайним користувачем та роботами, а також для різних типів атак, таких як міжсайтовий сценарій атаки, атаки з низькою швидкістю у DDOS.

Література

[8] Gupta S., Gupta B. B. Cross-Site Scripting (XSS) attacks and defense mechanisms: classification and state-of-the-art //International Journal of System Assurance Engineering and Management. – 2017. – Т. 8. – №. 1. – С. 512-530.

[9] MOH, Melody, et al. Detecting web attacks using multi-stage log analysis. In: *2016 IEEE 6th International Conference on Advanced Computing (IACC)*. IEEE, 2016. p. 733-738.

[10] Aliero, Muhammad Saidu, et al. "Classification of Sql Injection Detection And Prevention Measure." *IOSR Journal of Engineering* 6.02 (2016). – С. 75-93

[11] Shah, Nisharg. "Securing Database Users from the Threat of SQL Injection Attacks." (2017).

[12] Sriram A. et al. Cold fusion: Training seq2seq models together with language models //arXiv preprint arXiv:1708.06426. – 2017.

Дослідження відкритих сигнатурних систем виявлення аномалій

Владислав Труш, Андрій Сторожов, Олександр Федюшин

Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м. Харків, пр. Науки, 14, E-mail: vladyslav.trush@nure.ua, andrii.storozhov@nure.ua , oleksandr.fediushyn@nure.ua

***Коротка анотація** – A comparative analysis of the characteristics of modern open source anomaly detection systems was performed. The comparison this system was performed on the following indicators: monitoring level, response type, adaptive capacity, architecture, detection method supported for platform deployment, principle of construction, method of extending functionality.*

Ключові слова – системи виявлення аномалій, сигнатури, мережні атаки, Snort, Suricata, Bro, OSSEC, Prelude.

I. Вступ

В епоху цифрових технологій однієї із ключових проблем у сучасному суспільстві є захист інформації. Це приводить до створення певних систем і засобів, які повинні мати у своїй наявності на мережному рівні деякі механізми захисту й протидії хакерським загрозам. У цей час вважається важливим забезпечити найбільший рівень захищеності в корпоративних мережах будь-якої організації, яка зберігає критично важливі дані. Успіх і конкурентоспроможність такої організації багато в чому залежать саме від використання ефективних засобів, що задовольняють умовам вище.

Одним з найбільш відомих засобів, що добре зарекомендували себе, для розв'язання цього завдання є системи виявлення аномалій (СВА) [1]. По способу збору інформації виділяють хостові та мережні СВА. У даній роботі акцент зроблений убік дослідження та критичного аналізу тих СВА, які функціонують саме на мережному рівні. Такі СВА можна визначити як програмні або апаратні пристрої, призначені для виявлення спроб несанкціонованого доступу, фактів порушення безпеки в рамках мережної взаємодії між хостами та будь-яких інших аномальних дій у комп'ютерній мережі.

II. Аналіз проблеми

Для виявлення мережних атак можуть застосовуватися як сигнатурні механізми пошуку шаблонових аномальних дій, так і евристичні (статистичні, нейромережні, імунні та ін.) підходи. У випадку сигнатур [1-4] розв'язання завдання зводиться до реалізації процедури, що виконує перевірку входження заданої байтової послідовності усередині вмісту мережних пакетів. Недоліками такого розв'язку є складність створення репрезентативного набору з подібними записами й обмеження у виявленні модифікованих варіантів відомої атаки. Навпаки, евристичні підходи дозволяють виявляти сховані закономірності в аналізованих мережних потоках. Саме ця особливість пояснює їхню широку популярність у науково-дослідницькому співтоваристві й відіграє ключову роль при виборі й проектуванні ядра СВА. З іншого боку, в основі функціонування більшості комерційних і відкритих програмних рішень переважає підхід, який базується на сигнатурному зіставленні зі зразком і характеризується мінімальним числом неправильних спрацьовувань. Для збереження переваг обох підходів використовується прийом їх комбінування, який як і раніше залишається не повною мірою дослідженим. Тому завдання виявлення аномальних мережних з'єднань є актуальним.

III. Розв'язання проблеми

Для огляду були обрані п'ять СВА, які мають відкритий програмний код і поширюються по безкоштовних ліцензіях GNU GPL і BSD. Серед них представлені Snort, Suricata, Bro, OSSEC, Prelude [2,4]. Аналіз даних СВА проводився на рівні вивчення їх вихідних текстів, керівництв користувача, програмної документації й інших джерел технічної та наукової літератури, що перебувають у відкритому доступі.

У Табл. 1 представлені порівняльні характеристики і їх значення для кожної із представлених вище СВА.

Порівняння СВА проводилося за наступними показниками: рівень моніторингу, тип реагування, адаптивна здатність, архітектура, метод виявлення, підтримувані для розгортання платформи, принцип побудови, спосіб розширення функціональності.

Майже всі описані СВА є активними. Виключенням із цього списку є лише Bro, яка не має вбудованих засобів для запобігання атак, однак за допомогою модуля, що йде в поставці з нею, exec.bro можна настроїти примусове скидання підозрілого з'єднання або блокування трафіка на рівні ядра за допомогою правил IPtables. Це анітрошки не збіднює дану платформу, оскільки згідно із джерелом [1] Bro розроблялася в першу чергу саме для вивчення характеристик мережного трафіка, а не для виявлення в ньому атак. Із цією метою в ній були реалізовані потужні аналізатори протоколів, що дозволили ідентифікувати тип протоколу навіть на нестандартних портах завдяки механізму DPD (Data Packet Detection, Dynamic Protocol Detection), який аналогічний застосовуваному в Suricata. Крім того, при розробці Bro позначилося її минуле: вона розвивалася в академічному оточенні, тому характеризується властивістю адаптивності за рахунок наявності в її основі статистичних модулів виявлення мережних аномалій: спроб сканування портів (scan.bro), проведення Dos-атак (synflood.bro). Усі із представлених СВА є кроссплатформеними, за винятком Bro, яка призначена для запуску тільки в Unix-подібних ОС. Також відзначимо, що для

ТАБЛИЦЯ 1.

ПОРІВНЯЛЬНІ ХАРАКТЕРИСТИКИ СВА

Хар-ка	Snort	Suricata	Bro	Prelude	OSSEC
Рівень моніторингу	Мережний	Мережний	Мережний	Мережний та хостовий	Хостовий
Тип реагування	Активний	Активний	Пасивний	Активний	Активний
Здатність до адаптації	Додатковий модуль	Додатковий модуль	Статистичний аналіз	Відсутній	Відсутній
Архітектура	Централізована	Централізована	Централізована	Розподілена	Розподілена
Платформи	Windows, *nix	Windows, *nix	*nix	Windows, *nix	Windows, *nix
Спосіб розширення	Динамічно завантажувані декодери	Динамічно завантажувані so/dll бібліотеки, Lua-скрипти	Bro-сценарії, so-бібліотеки	Сенсори, що підключаються	XML-правила

платформи Windows система OSSEC не має режиму локальної установки, тобто для цієї ОС необхідно встановлювати окремо й сервер, і агенти. І в той же час саме OSSEC є єдиною із представлених СВА, здатною виявляти шкідливі процеси як наслідки успішно реалізованих атак. Для систем Snort і Suricata властивість адаптивності найчастіше реалізується через додаткові модулі. Зустрічаються як вітчизняні, так і закордонні дослідження, у яких автори вбудовують в Snort такі інтелектуальні декодери, приміром, на основі нейронних мереж, методу опорних векторів і дерев розв'язків. За принципом побудови можна класифікувати дані системи як монолітні та компонентні. У першому випадку система представлена як єдиний бінарний файл, запуск якого засобами ОС, як правило, породжує не більш одного процесу. До них з розглянутих СВА належать Snort, Suricata і Bro. Компонентний підхід має на увазі розбивку системи на кілька функціональних блоків, кожний з яких запускається в окремому просторі пулу адрес пам'яті, виконує конкретне завдання й спілкується з іншими на основі механізмів міжпроцесної взаємодії. До ряду таких систем належать OSSEC і Prelude. Як уже було відзначено раніше, основними елементами для створення додаткових функцій у роботі систем Snort і Suricata є декодери й препроцесори, основа функціонування яких полягає у виклику певних функцій з динамічно завантажуваних бібліотек (so/dll). Найбільш просунутий спосіб створення нових модулів, що розширюють функціональні можливості системи, має Bro. Уся рутинна робота з написання шаблонових файлів майбутнього додатка зведена до мінімуму. Разом з вихідними кодами даної системи поставляється bash-скрипт init-plugin, призначений для автоматичної генерації початкового кістяка майбутнього модуля, правил його складання й установки. Після компіляції модуль являє собою готову so-бібліотеку, прототипи функцій з якої обгорнені у відповідний bro-скрипт. Наступне завантаження, ініціалізація покажчиків на функції та деалокація модулів системою зводяться до виклику функцій dlopen, dlsym і dlclose відповідно.

Висновки

В роботі проведений порівняльний аналіз характеристик сучасних відкритих систем виявлення аномалій.

Серед розглянутих СВА найбільш повні характеристики має Prelude. Будучи спроектованою з самого початку розподіленою системою, вона підтримує гібридний моніторинг контрольованих вузлів, здійснюючи аналіз як на рівні мережі, так і на рівні хоста. Крім того, ця система є масштабованою, що дозволяє їй використовувати безліч різнорідних джерел для збору й обробки даних. Модульний принцип установки цієї системи також дозволяє добитися більш гнучкого настроювання кожного з її компонентів окремо. Можливість підключення кожної із представлених СВА в якості сенсорів для Prelude (для Bro необхідно реалізовувати нового клієнта вручну, тому що для неї немає готового сенсора) ставить її на ранг вище інших СВА.

Література

[1] Rødfoss, J. T. Comparison of open source network intrusion detection systems / J. T. Rødfoss: MA thesis / Rødfoss, Jonas Taftø. — University of Oslo, Department of Informatics, 2011. — 85 pp.

[2] Sanders, C. Applied network security monitoring: collection, detection, and analysis / C. Sanders, J. Smith. — Elsevier, 2013. — 496 pp.

[3] ntop. Accelerating Snort, Bro and Suricata with PF RING ZC [Електронний ресурс]. – URL: http://www.ntop.org/pf ring/acceleratingsnort-bro-and-suricata-with-pf ring-zc/

[4] Zaraska, K. Prelude IDS: current state and development perspectives / K. Zaraska [Електронний ресурс]. – 2003. – URL: http://citeseerx.ist.psu.du/viewdoc/download?doi=10.1.1.106.5542%5C&rep=rep1%5C&type=pdf.

Проблеми оперативного виявлення і реагування на інциденти інформаційної безпеки

Владислав Ушатов[1], Олександр Сєвєрінов[2]

1. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м Харків, пр. Науки, 14, E-mail: vladyslav.ushatov@nure.ua

2. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м Харків, пр. Науки, 14, E-mail: oleksandr.sievierinov@nure.ua

Коротка анотація – The issues of the effective use of information protection and event management systems are considered. The analysis of the problems arising when using these systems in organizations is carried out. Possible solutions to these problems used in the latest versions of SIEM systems are considered.

Ключові слова - управління інформаційною безпекою, подіями безпеки, інциденти інформаційної безпеки, SIEM.

I. Вступ

Розвиток інформаційних технологій призводить до збільшення випадків витоку інформації. Результати глобального дослідження компанії InfoWatch витоків конфіденційної інформації в першому півріччі 2019 року показали, що за даний період аналітиками було зареєстровано тисяча двісті сімдесят шість випадків витоку конфіденційної інформації, з яких 55,6% відбулися в результаті внутрішніх порушень, а 44,4% через зовнішній вплив [1]. Сукупна кількість скомпрометованих призначених для користувача даних перевищила показник першого півріччя 2018 року більш ніж в 3,6 рази і склала 8,74 млрд записів. На рисунку 1 представлений зріст числа зареєстрованих випадків витоку інформації за останні 12 років (прогноз на 2019 рік за підсумком результатів першого півріччя).

В цих умовах, а також враховуючи вимоги міжнародних стандартів, однією з важливих засобів захисту інформації стає система управління інцидентами інформаційної безпеки [2].

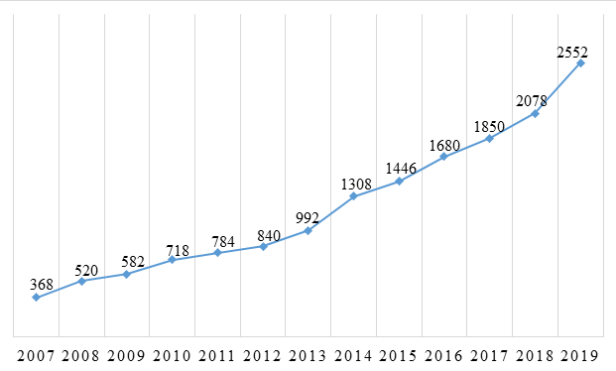


Рисунок 1 - Число зареєстрованих витоків інформації

Тому актуальним є проведення аналізу проблеми оперативного виявлення і реагування на інциденти інформаційної безпеки.

II. SIEM - системи управління інформаційною безпекою та подіями безпеки

Незалежно від кількості вже проваджених засобів захисту в організації, якщо своєчасно не реагувати на виникаючі загрози інформаційної безпеки (ІБ), то їх ефективність буде прагнути до нуля. При цьому з ростом обсягів інформаційних потоків стрімко зростає і кількість засобів ІБ, від яких надходять дані про події інформаційної безпеки, тому встежити за рівнем безпеки організації стає все складніше і складніше.

Через величезні обсяги оброблюваних даних стає складно сфокусуватися на важливих аспектах інформаційної безпеки підприємства. Тому для оперативного виявлення і реагування на інциденти ІБ в сучасних великих підприємствах використовуються рішення класу SIEM (Security Information and Event Management, системи управління інформаційною безпекою та подіями безпеки) [3, 4] .

Функціональна модель системи SIEM об'єднує підсистеми: збору даних, попередньої їх обробки, зберігання, аналізу, уявлення. Виходячи з цього узагальнена послідовність обробки подій безпеки в рішеннях SIEM представлена на рисунку 2 [3].

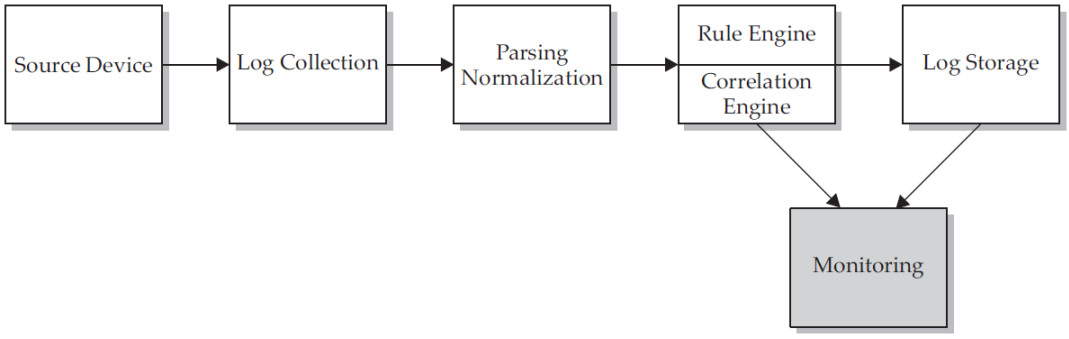


Рисунок 2 - Узагальнена послідовність обробки подій безпеки в системах SIEM

Застосовувані методи і засоби в системах SIEM розділити по функціональності на наступні основні групи: збір і агрегація (об'єднання елементів); аналіз і кореляція (знаходження взаємозв'язку); оповіщення; візуалізація; зберігання; експертний аналіз і пошук; допоміжні методи і засоби.

Системи SIEM забезпечують аналіз в реальному часі подій (інцидентів) безпеки, отриманих від мережевих пристроїв і додатків.

Через високу складність системи SIEM мають велику кількість проблем, що заважають їх ефективному застосуванню.

III. Проблеми застосування систем SIEM

Проведений аналіз показав, що одним з перших недоліків систем управління інформаційною безпекою та подіями безпеки, що заважає широкому використанню їх в малому і середньому бізнесі, є висока вартість.

Другим проблемним питанням є те, що через складність і високі вимоги системи безпеки SIEM часто не виправдовують очікування керівництва організацій і користувачів, і мета їх використання не досягається за рік і більше. За даними Gartner, компанії часто купують друге або третє SIEM-рішення, тому що чинне рішення не виправдало очікувань [5]. Відсутність необхідних ресурсів і навичок обслуговуючого персоналу (фахівців з досвідом з розслідування інцидентів, аудиту і тестів на проникнення) - найпоширеніші причини невдалих SIEM-проектів.

Наступними э низька проблем, пов'язаних з технічними питаннями застосування систем управління інформаційною безпекою та подіями безпеки. Однією з причин низької ефективності використання SIEM-систем є те, що необхідно постійно змінювати налаштування системи - для того щоб вона продовжувала збирати актуальні дані з джерел і виявляти інциденти. Ще однією проблемою – є непрацюючі правила кореляції. В інфраструктурі організації постійно відбуваються конфігураційні зміни (зміна структури мережі, підключення нових засобів, оновлення продуктів), що призводять до недійсності правил, помилкових спрацьовувань або пропуску безлічі подій. Зі змінами в інформаційній структурі доводиться додавати нові правила нормалізації, кореляції і агрегації в систему SIEM. І не завжди фахівці організації це можуть зробити самостійно – не мають досвіду або система SIEM не має такого функціоналу.

Проблемним питанням для більшості організацій є також трудовитрати на роботу з SIEM-системою. Найчастіше розмір персоналу, що обслуговує систему SIEM складає від двох до п'яти осіб, а їх час роботи з системою постійно зростає.

IV. Шляхи вирішення проблем застосування SIEM-систем

Дослідження компанії Positive Technologies показало, що на думку респондентів, знизити трудовитрати на SIEM-систему допоможуть: поставка вендором способів детектування загроз (53%); керівництво по донастройці правил для зниження кількості помилкових спрацьовувань (49%); можливість писати власні правила кореляції без вивчення спеціальної мови (44%).

Деякі розробники SIEM-систем впроваджують ці побажання в своїх виробах. Нові версії SIEM-систем мають в своєму функціоналі конструктор правил кореляції, за допомогою якого фахівці організації можуть створювати власні правила, не використовуючи будь-якої спеціальної мови програмування.

В системах SIEM з'явилася можливість проводити ретроспективний аналіз за індикаторами компрометації, що дозволяє виявляти атаки, що сталися в минулому, і запобігти їх подальшому розвитку.

Для виявлення нових загроз компанія Positive Technologies запропонувала хмарну базу знань Positive Technologies Knowledge Base (PT KB). Вона автоматично наповнює підключені SIEM новими способами виявлення компрометації в вигляді правил кореляції, агрегації, нормалізації, а також інформації про способи розслідування інцидентів.

Також останнім часом в функціонал SIEM-систем має можливості отримувати актуальні дані про стан ІБ у великій організації в будь-який момент і виявляти розподілені атаки на інфраструктуру окремого підрозділу або цілого підприємства.

Висновки

Таким чином, незважаючи на достатню кількість проблем застосування SIEM-систем ведеться активна робота по їх модернізації, що дозволяє організаціям ефективно використовувати ці системи для оперативного виявлення і реагування на інциденти інформаційної безпеки.

Література

[1] Сайт компанії InfoWatch [Електронний ресурс]. – Режим доступу: https://infowatch.com.

[2] Сєвєрінов О.В. Управління інформаційною безпекою згідно міжнародних стандартів / О.В. Сєвєрінов, В.І. Черниш, М.Є. Молчанова // Системи управління, навігації та зв'язку. – К: ДП «ЦНДІ НіУ». - 2011. – Вип. 4(20). – С. 250-253.

[3] Miller D. et al. Security information and event management (SIEM) implementation. – McGraw-Hill, 2011.

[4] Сєвєрінов О.В. Управління інцидентами інформаційної безпеки на основі використання SIEM систем / О.В. Севєрінов, В.В. Ушатов // Інформатика, управління та штучний інтелект. Тези шостої міжнародної науково-технічної конференції – Х.: НТУ «ХПІ», 2019. – С. 109.

[5] Сайт компанії Gartner [Електронний ресурс]. – Режим доступу: https://www.gartner.com/.

[6] Трудозатраты на работу в SIEM выросли у 62% специалистов по ИБ [Електронний ресурс]. – Режим доступу: https://ko.com.ua/taxonomy/term/8247.

Загрози в мережі Blockchain

Дмитро Фесенко[1], Володимир Караваєв[2]

1. Кафедра безпеки інформаційних технологій,
Харківський національний університет радіоелектроніки,
УКРАЇНА, г. Харків, пр. Науки, 14,
E-mail: dmytro.fesenko@nure.ua

2. Кафедра безпеки інформаційних технологій,
Харківський національний університет радіоелектроніки,
УКРАЇНА, г. Харків, пр. Науки, 14,
E-mail: volodymyr.karavaiev@nure.ua

Коротка анотація – Blockchain is a digital system that is based on model without a central storage. To design the system according to the requirements of blockchain, developers must take to attention rules on using such kind of system. The only system's thread itself that will be implemented on the blockchain during development, will make your system much less robust, so to create a really good-secured system developers should understand all threads in blockchain and to be possible to persist them.

Ключові слова – Blockchain, Атака, Вразливість, Децентралізація, Підміна.

I. Вступ

Ідеологія проектів, що застосовують технологію блокчейн - це можливість розмежованого використання доступу до інформації з унеможливленням зміни даних в реєстрі, що доступний усім користувачам мережі. Ця технологія стала відома коли вона була застосована для забезпечення появи електронних грошей, де здійснюються цифрові перекази грошей у розподілених системах. Через це блокчейн часто розглядається як пов'язаний з біткоїном або рішеннями в сфері електронної валюти в цілому. Незважаючи на це, технологія може бути використана більш широко для різних сфер застосування.

II.Побудова систем з використанням blockchain

Кожен компонент блокчейн можна просто описати та використовувати як структурний елемент, щоб зрозуміти більш складну систему.

В системі до старих даних додаються нові блоки, попередні блоки стає все важче модифікувати. Нові блоки додаються до всіх копій учасників реєстру в мережі та будь-які конфлікти вирішуються автоматично за допомогою встановлених правил.

Системи блокчейн можуть здаватися складними, але їх можна легко зрозуміти вивчаючи кожну компонентну технологію індивідуально.

Важливим компонентом технології блокчейн є використання криптографічних геш-функцій для гешування вмісту блоку, наприклад, в багатьох технологіях пов'язаних з блокчейн використовується алгоритм гешування Secure Hash Algorithm (SHA) з вихідним розміром 256 біт. Багато комп'ютерів підтримують цей алгоритм в апаратному забезпеченні, що робить його швидким для обчислення.

Використаний алгоритм гешування (SHA-256) є стійким до колізій (пошуку двох однакових дайджестів для різних початкових значень), тому що для виявлення такої ситуації в SHA-256 потрібно було б виконати алгоритм в середньому близько 2128 разів. Технології блокчейн беруть список транзакцій та створюють геш для списку. Кожен, хто має такий же список транзакцій, може генерувати точно такий самий відбиток. Якщо змінюється одне значення транзакції в списку, дайджест для цього блоку змінюється, що дозволяє легко виявити навіть незначні зміни в одному біті [1].

Важливо визначити дійсність транзакції тому, що те, що хтось стверджує, що транзакція відбулася, не означає, що це дійсно відбулося. Транзакції підписані, та їх можна будь-коли перевірити за допомогою пари публічних / приватних ключів.

Для передачі цінності в мережах блокчейн використовуються адреси.

Адреса користувача – це короткий буквено-цифровий рядок, що походить від відкритого (публічного) ключа користувача, використовуючи функцію гешування. Адреси використовуються для надсилання та отримання цифрових активів. Адреси коротші за відкриті ключі та не таємні. Для створення адреси необхідно взяти відкритий ключ, його геш-значення та перетворення гешу в текст.

Користувачі можуть генерувати так багато пар приватних / публічних ключів, а значить, адрес, як того хочуть, дозволяючи змінювати ступінь псевдоанонімності. Адреси виступають як ідентифікація блокчейну для користувача та часто за допомогою простого користування адреса перетворюється на QR-код. Підписавши цифрову транзакцію за допомогою приватного ключа, транзакцію можна перевірити за допомогою публічного ключа.

В блокчейні використовується асиметрична ключова криптографія, що використовує пару ключів: публічний та приватний ключ, які математично пов'язані один з одним. Публічний ключ може бути оприлюднений без зниження безпеки процесу, але приватний ключ повинен залишатись секретом, якщо інформація має зберігати свій криптографічний захист. Попри те, що існує взаємозв'язок між двома ключами, приватний ключ не може ефективно визначатися на підставі знання публічного ключа.

Під час роботи кожного вузла мережі блокчейн використовується та зберігається реєстр обліку, що є сукупністю транзакцій, які передавалися між усіма вузлами за весь час роботи [1].

Учасники мережі можуть подавати кандидати транзакцій в реєстр обліку, відправивши ці транзакції до деяких вузлів, що беруть участь у блокчейні. Подані транзакції поширюються на інші вузли мережі

(але це само по собі не включає транзакцію в блокчейн). Розподілені транзакції очікують у черзі або пулі транзакцій, доки вони не будуть додані до блокчейну вузлом видобування (процес майнінгу).

Після створення кожний блок гешується, створюючи таким чином дайджест, що представляє блок. Зміна навіть одного біта в блоці повністю змінить геш-значення. Дайджест блоку використовується для захисту блоку від змін, після того, як всі вузли матимуть копію геш-значення блоку, можна перевірити, чи не було змінено блок.

III.Атаки на мережу блокчейн

Стосовно мережі блокчейн, порушники можуть бути зовнішніми або внутрішніми [2]. Внутрішній порушник – це, здебільшого, розробник певного проекту або додаткового компоненту (модуля) проекту, що хоче порушити безпеку системи ще на етапі розробки або експлуатації, реалізує внутрішнє проникнення або віддалене, де використовується вже відома помилка у безпеці. Зовнішній – атакує мережу, як звичайний користувач або учасник мережі (вузол або нод мережі), реалізує віддалені атаки.

Мета порушника:

1) одержання можливості вносити зміни в мережу блокчейн згідно зі своїми намірами;

2) перешкоджати нормальній роботі мережі (перевантажувати вузли, блокувати доступ, отримати контроль над мережею тощо);

3) отримання матеріальної або іншої вигоди, шляхом крадіжки даних / грошових ресурсів.

Для того, щоб здійснити атаку, внутрішній порушник повинен:

1) мати доступ до редагування програмної частини мережі або створення та додавання до неї власних частин (бекдорів), що дадуть змогу використати мережу у своїх цілях;

2) володіти достатніми знаннями про роботу мережі та про її вразливі місця.

Зовнішній порушник повинен:

1) володіти достатньою кількістю обчислювальних та матеріальних ресурсів для здійснення своєї атаки;

2) володіти достатніми знаннями про роботу мережі та про її вразливі місця.

У ситуації для внутрішнього порушника, технічна оснащеність не є важливою тому, що використання бекдорів або заздалегідь створених каналів для несанкціонованого доступу є простою здачею для одного звичайного комп'ютера.

У ситуації для зовнішнього порушника, ресурси для атаки є вкрай важливими через те, що безпека багатьох сучасних блокчейн систем опирається на складність вирішення певних криптографічних задач. Успішність атаки залежить від кількості обчислювальних ресурсів / потужностей зловмисника, якщо мова не йде про використання знайденої помилки у мережі безпеки [3].

Кількість людей, що здійснюють атаку, може бути різною і залежить лише від плану самої атаки. Для внутрішнього зловмисника достатньо і однієї людини. Атаку із необхідністю великої кількості ресурсів, одній людині буде виконати вже важче. Загалом, сучасні атаки на блокчейн або на користувачів мереж блокчейн, здійснюються підготовленою командою із продуманим планом дій на кожному етапі атаки.

В нашому випадку при розгляді безпеки ФГ, необхідно брати до уваги як внутрішніх, та і зовнішніх зловмисників. Щодо функцій гешування може бути застосована загроза «зламування криптоалгоритмів», в основі якої лежить необхідність в великій обчислювальній потужності.

Висновки

Кожен учасник мережі повинен використовувати шифрування (здебільшого асиметричне). Заходи безпеки вбудовані в мережу і закладені у загальному протоколі, що забезпечує конфіденційність і автентичність для користувачів.

Користувачі повинні контролювати свої дані. У кожного має бути право вирішувати, які відомості, коли, як і в якому обсязі повідомляти про себе – тобто ти сам володієш своїми конфіденційними даними і без твого відома їх ніхто не може отримати.

Права учасників системи – рівні, прозорі і закріплені та підтверджуються в мережі усіма і всі з ним погоджуються.

При виконанні всіх рекомендацій з безпеки, використання перевірених та надійних криптографічних засобів можна легко побудувати гарно захищену систему на основі технології блокчейн.

Література

[1] Dylan Yaga, Peter Mell, Nik Roby, Karen Scarfone. NISTIR 8202 Blockchain Technology Overview.

[2] Halpin H., Piekarska M. Introduction to Security and Privacy on the Blockchain //2017 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW). – IEEE, 2017. – С. 1-3.

[3] Tosh D. K. et al. Security implications of blockchain cloud with analysis of block withholding attack //Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing. – IEEE Press, 2017. – С. 458-467.

Обґрунтування стійкості алгоритму ЕЦП FALCON

Денис Черниш[1], Антон Янко[2]

1. Кафедра безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, УКРАЇНА, м. Харків, пр. Науки, 14,

2. Кафедра автоматизації проектування обчислювальної техніки, Харківський національний університет радіоелектроніки, УКРАЇНА, м. Харків, пр. Науки, 14, E-mail: denys.chernysh@nure.ua

Коротка анотація – This article discusses the known cryptographic attacks on the electronic digital signature algorithm FALCON. We also consider such a weak point in the algorithm as floating point operations. We have considered the attacks that are most effective among existing attacks on electronic digital signature algorithms built on algebraic lattices. For the FALCON EDS algorithm, these attacks are ineffective when using the parameters given in the paper. We have also provided a justification for the floating point arithmetic algorithm used in the algorithm, which allows the algorithm to be immune to attacks related to this problem. Other algorithm vulnerabilities are currently being investigated within the NIST PQC competition.

Ключові слова – алгоритм FALCON, решітки, NTRU, швидке перетворення фур'є, гешування, електронний цифровий підпис, семплер, структура GPV, постквантова криптографія

V. Вступ

Криптосистеми на основі алгебраїчних решіток є перспективним напрямком постквантової криптографії. Основним класом решіток що використовуються у таких криптосистемх є решітки NTRU. Використання цих решіток дозволяє зробити алгоритм не лише швидким у порівнянні з іншими класами решіток, а й забезпечити порівняно високий рівень стійкості. Окрім цього, алгоритми електронного цифрового підпису засновані на алгебраїчних решітках мають порівняно малий розміру підпису та ключів. В даній роботі розглядається алгоритм ЕЦП FALCON[1] заснований саме на алгебраїчних решітках NTRU. В FALCON стійкість необхідний рівень стійкості забезпечується використанням NTRU-решіток у купі з структурою генерування цифрових підписів GPV.

Метою роботи є дослідження відомих атак на алгоритм ЕЦП FALCON та дослідження стійкості даного алгоритму.

VI. Відомі атаки

Відновлення ключів. Найбільш ефективні атаки походять від зменшення решітки. Почнемо з розгляду решітки $\left(\mathbb{Z}[x]/(\phi)\right)^2\begin{bmatrix}0 & q\\ 1 & h\end{bmatrix}$. Після використання на цій основі редукції решітки перерахуємо всі точки решітки в області радіуса $\sqrt{2n}\sigma'$ з центром в початку координат. Таким чином, зі значною ймовірністю можна знайти $[g\ f]$. При використанні розміру блоку B, перерахування займає незначний час, якщо норма Грамма-Шмідта більше $0.75\sqrt{B}\sigma'$. Для найвідомішого алгоритму зменшення решітки, DBKZ, це $\left(\frac{B}{2\pi\mathrm{e}}\right)^{(1-n/B)}\sqrt{q}$.

Тоді легко вивести B і показати, що $B=n+o(n)$. Це дає $B=652$, при $n=768$, і $B=921$, при $n=1024$. Передбачувана безпека докладно описана в таблиці 1 з використанням методології New Hope [2].

ТАБЛИЦЯ 1

ПЕРЕДБАЧУВАНА БЕЗПЕКА

n	B	Класичний	Квантовий
512	392	114	103
768	652	195	172
1024	921	263	230

Підробка підпису. Підробка підпису може бути здійснена шляхом знаходження точки решітки на відстані, обмеженій β від випадкової точки, в тій же решітці, що і вище. Це завдання також полегшується завдяки першому виконанню редукції решітки на вихідній основі. Одна можливість полягає в тому, щоб перерахувати всі точки решітки в кулі радіусом $\sqrt{\frac{nq}{\pi\mathrm{e}}}$. Оскільки ця куля більше, ніж в попередній атаці, ця атака буде повільніше. Може здатися, що це буде повільніше, ніж попередня атака через фактор $\Theta\left(\sqrt{n}\right)$ в радіусі. Це не той випадок, оскільки решітка має ортогональний базис, з чого випливає, що на відстані в $o\left(\sqrt{n}\right)$ мало точок $\left(2^{o(n)}\right)$. Мається на увазі, що запропонований спосіб починається з відновлення секретного ключа, так що він повільніше, ніж попередній алгоритм. Крім того, вкладення точки в решітку не допомагає: відстань до решітки $\Theta\left(\sqrt{n}\right)$ більше, ніж найкоротша ненульова точка.

Комбінаторна атака. При $q=O(n)$, розмір коефіцієнтів був би постійним. Тоді варіант Кирхнера-Фуке [3] BKW буде відпрацьовувати за час $2^{n/((2+o(1))\log\log n)}$, щоб відновити ключ, тобто асимптотично швидше, ніж попередні алгоритми. Це вказує на те, що найбільш компактна схема використовує $q=n^{1+\varepsilon+o(1)}$ для деякого $\varepsilon>0$. Однак, оскільки n не велике, використаного q досить, щоб зробити цю атаку неактуальною. Дійсно, навіть якщо припустити, що пошук найближчого сусіда виконується за постійний час і інші оптимістичні припущення, найкраща комбінаторна атака виконується за час 2^{135} для $n=512$.

Гібридна атака. Гібридна атака [4] поєднує в собі алгоритм зустрічі посередині і алгоритм відновлення ключа. Це дуже ефективно використовувалося проти NTRU, через його вибір розріджених многочленів. Це,

однак, не той випадок, так що його вплив набагато скромніший і врівноважується відсутністю сітчастого перерахування.

Щільна підрешітка високого рангу. Роботи показали, що, коли f,g надзвичайно малі в порівнянні з q, легко атакувати криптографічні схеми, засновані на решітках NTRU. Навпаки, в FALCON f,g приймаються не дуже малими, в той час як q невеликий: побічний ефект полягає в тому, що це робить схему непроникною для так званих атак «надмірного навантаження NTRU». Зокрема, навіть якби f,g були взяті в двійковому вигляді, довелося б обрати $q > n^{2.83}$, щоб ця властивість була корисною для криптоаналізу.

VII. Точність арифметики з плаваючою точкою

Семплер з пасткою зазвичай вимагає використання арифметики з плаваючою точкою, і швидкий семплер Фур'є не є винятком. Це ставить питання про точність, необхідної для встановлення значущих кордонів безпеки. Наївний аналіз зажадав би точності $O(\lambda)$ бітів (незважаючи на логарифмічні фактори), але це привело б до істотно більш повільної процедурі генерації підпису.

Щоб проаналізувати необхідну точність, використовується аргумент розбіжності Рені. Як і в [5], через $a \prec b$ позначається той факт, що $a \le b + o(b)$, що дозволяє строго відкидати незначні фактори. Представлений швидкий семплер Фур'є є рекурсивним алгоритмом, заснованим на $2n$ дискретних семплерах $D_{\mathbb{Z},c_j,\sigma_j}$. Припускається, що значення c_j (відповідно σ_j) відомі з абсолютною помилкою (відповідно, відносною похибкою), що не перевищує δ_c (відповідно δ_σ), і через Υ (відповідно $\overline{\Upsilon}$) позначається вихідний розподіл семплера з нескінченної точністю. Потім стає можливим повторно використовувати точний аналіз семплера Кляйна. Для будь-якого виходу використаного семплера в гіршому випадку:

$$\left|\log\left(\frac{\overline{\Upsilon}(z)}{\Upsilon(z)}\right)\right| \prec 2n\left[\frac{\sqrt{154}}{1.312}\delta_c + (2\pi+1)\delta_\sigma\right] \le 20n(\delta_c+\delta_\sigma) \qquad (1)$$

В середньому випадку значення $2n$ в рівнянні 1 можна замінити на $\sqrt{2}n$. Дотримуючись аргументів безпеки, це дозволяє стверджувати, що в середньому не очікується втрат безпеки, якщо $(\delta_c+\delta_c) \le 2^{-46}$.

Щоб перевірити, чи так це для FALCON, FALCON запускався з двома різними значеннями точності, високою точністю 200 біт і стандартної точністю 53 біта, і порівнювалися значення c_j, σ_j. Результатом цих експериментів є те, що завжди є $(\delta_c+\delta_c) \le 2^{-40}$: хоча це більше, ніж 2^{-46}, різниця становить всього 6 біт. Тому можна вважати, що 53 біта точності досить для параметрів: рівень безпеки $\lambda \le 256$, кількість запитів $q_s \le 2^{64}$, і що можливість процедури підпису з витоком інформації про секретний базисі є суто теоретичною загрозою.

VIII. Висновки

Розглянуті атаки є найефективнішими серед існуючих атак на алгоритми електронного цифрового підпису побудованих на алгебраїчних решітках. Для алгоритму ЕЦП FALCON наведені атаки є неефективними при використанні наведених у роботі параметрів. Також наведено обґрунтування використовуваної у алгоритмі точності арифметики з плаваючою точкою яка дозволяє алгоритму бути невразливим до атак пов'язаних з цією проблемою. Пошук інших вразливостей алгоритму на даний момент проводить у рамках конкурсу NIST PQC.

Література

[1] Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU. Specifications v1.0. Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Prest, Thomas Ricosset, Gregor Seiler, William Whyte, Zhenfei Zhang. https://falcon-sign.info/falcon.pdf.

[2] Erdem Alkim, Léo Ducas, Thomas Pöppelmann, and Peter Schwabe. Post-quantum key exchange − A new hope. In 25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016., pages 327–343, 2016.

[3] Paul Kirchner and Pierre-Alain Fouque. An improved BKW algorithm for LWE with applications to cryptography and lattices. In Rosario Gennaro and Matthew J. B. Robshaw, editors, CRYPTO 2015, Part I, volume 9215 of LNCS, pages 43–62, Santa Barbara, CA, USA, August 16–20, 2015. Springer, Heidelberg, Germany.

[4] Nick Howgrave-Graham. A hybrid lattice-reduction and meet-in-the-middle attack against NTRU. In Alfred Menezes, editor, CRYPTO 2007, volume 4622 of LNCS, pages 150–169, Santa Barbara, CA, USA, August 19–23, 2007. Springer, Heidelberg, Germany.

[5] Erdem Alkim, Léo Ducas, Thomas Pöppelmann, and Peter Schwabe. Post-quantum key exchange – A new hope. In 25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016., pages 327–343, 2016.

АЛФАВІТНИЙ ПОКАЖЧИК

ЗМІСТ

Матеріали першого міжнародного науково-практичного форуму

Відповідальний випусковий Г.З. Халімов
В.В. Лебідь

Комп’ютерна верстка О.В. Сєвєрінов

Матеріали збірника публікуються в авторському варіанті без редагування

Затверджено Науково-технічною радою Харківського національного університету радіоелектроніки №11 від 16.11.2019 р.