КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 2 (10), 2020
ISSN 2663 - 4023
DOI 10.28925/2663-4023.2020.10.628
УДК 621.3.019.3+004.056

Гулак Геннадій Миколайович
к.т.н., доцент, завідувач лабораторії науково-дослідний відділ №235
Інститут проблем математичних машин і систем, Київ, Україна
ORCID: 0000-0001-9131-9233
h.hulak@ukr.net

Бурячок Володимир Леонідович
д.т.н., професор, завідувач кафедри інформаційної та кібернетичної безпеки
Київський університет імені Бориса Грінченка, Київ, Україна
ORCID: 0000-0002-4055-1494
v.buriachok@kubg.edu.ua

Складанний Павло Миколайович
старший викладач кафедри інформаційної та кібернетичної безпеки
Київський університет імені Бориса Грінченка, Київ, Україна
ORCID: 0000-0002-7775-6039
p.skladannyi@kubg.edu.ua

Кузьменко Лідія Володимирівна
методист ІІ категорії центру перспективного планування та моніторингу освітньої діяльності
Національної академії Служби безпеки України, Київ
ORCID: 0000-0001-7392-0324
lido4ok@gmail.com

КРИПТОВІРОЛОГІЯ: ЗАГРОЗИ БЕЗПЕКИ ГАРАНТОЗДАТНИМ ІНФОРМАЦІЙНИМ СИСТЕМАМ І ЗАХОДИ ПРОТИДІЇ ШИФРУВАЛЬНИМ ВІРУСАМ

Анотація В даній роботі розглянуто загрози безпеки гарантоздатними інформаційним системам, а також сформовані заходи щодо протидії шифрувальним вірусам. Визначена типова послідовність кібератак з шифруванням інформації за допомогою програмних засобів реалізації атак. Описана послідовність процедур шифрувальної частини WannaCry. В роботі запропонована характеристика обчислювальної складності задач відновлення зашифрованих даних, зокрема виділені методи розпаралелювання розв'язку задач криптоаналізу, методи, що використовуються для розв'язку задач оцінювання стійкості криптосистем, пошуку вразливості та дешифрування залежно від базових математичних методів. Описано застосування технології розподілених обчислень для вирішення задач відновлення зашифрованих ресурсів. В роботі констатовано, що в сучасних умовах постійного розвитку методів криптографії з високім рівнем стійкості та їх широкою доступністю необхідною умовою підвищення ефективності відновлення зашифрованих програмами – вимагачами даних є створення спеціального програмного монітору безпеки та побудови спеціалізованих багатопроцесорних систем для реалізації методів криптоаналізу із широким доступом авторизованих користувачів, з точки зору мінімізація співвідношення "вартість - отриманий результат" найбільш раціональним підходом до створення спеціалізованих багатопроцесорних систем є побудова кластерної системи на базі найбільш потужних комп'ютерів загального призначення із застосуванням апаратних прискорювачів обчислень на базі програмованих логічних інтегральних схем, для підвищення ефективності атак на криптографічні програмні засоби реалізації атак доцільно розвивати технології активних дій у кібернетичному просторі, зокрема, такі, що забезпечують утворення прихованих каналів.

Ключові слова: криптовірологія, гарантоздатні інформаційні системи, кібератака, шифрування, криптографія, криптоаналіз, криптоалгоритм, програмовані логічні інтегральні схеми, програмні засоби реалізації атак

1. ВСТУП

Криптологія з початку її застосування і майже до останнього часу переважно відповідала на наступні два базових питання. З точки зору сторони, що убезпечує власний ресурс: Як забезпечити надійний захист інформації від сторонніх осіб шляхом її криптографічних перетворень? Атакуюча сторона шукала відповідь: Яким чином подолати захисні бар'єри за для досягнення власної мети?

Атаки на інформаційні системи, які спостерігались протягом останнього часу застосовували віруси, які шифрували інформаційні ресурси зазначених систем, включаючи бази даних, окремі текстові, аудіо і відео файли, сформулювали третє питання. Як відновити зашифровані дані за оперативне придатний час? Хоча перші результати науково-практичних досліджень у галузі криптовірології були отримані ще у 90-х роках минулого століття [1], та були продовжені у роботах [2-6]. Значний внесок у справу дослідження проблем зловмисного шифрування внесли Адам Юнг (A. Young), Моти Янг (M. Yung), Метт Суше (Matthieu Suiche) та інші. У той же час, завчасно практичні спеціалізовані рекомендації щодо протидії шифрувальним вірусам або щодо відновлення зашифрованих файлів отримані не були. Можливо вважати, що величезні фінансові втрати в світі внаслідок атак криптографічних вірусів у 2016-2017 р.р. були фактично запрограмовані недостатніми дослідженнями вказаної проблеми.

Зауважимо, що переважна більшість наукових робіт у сфері криптоаналізу на поточний час присвячена теоретичним аспектам побудови атак на основі певних вихідних даних (наприклад, на основі відомого шифротексту, відомих відкритих текстів й відповідних шифротекстів тощо). На сьогодні існує доволі значна кількість наукових публікацій, присвячених зазначеній вище проблематиці. Найбільш відомими серед них є роботи Е.Біхама, А.Шамира, М.Мацуи, Р.Рюппеля, Б.Шнайера, вчених радянської та вітчизняної математичних школ І.М. Коваленка, І.Д. Горбенка, А.В. Бабаша, Г.П. Шанкина та інших фахівців [7-11], в яких розглядається сучасний науково-методичний апарат методів дешифрування.

Слід згадати про дослідження, що присвячені атакам на реалізацію, тобто визначенню властивостей конкретних типів засобів криптографічного захисту інформації (КЗІ), особливості побудови яких дають можливість пасивно або активно (з впливом на засіб) реалізовувати деякий алгоритм повного або часткового дешифрування. Найбільш відомими серед відкрито опублікованих є роботи Грушо О.О., Роланд С. (Rowland C. H.), Келси (Kelsey J.) та інших дослідників [12-14]. Таким чином, аналіз публікацій у предметній області, що розглядається, свідчить про відсутність комплексного дослідження щодо проблеми підвищення ефективності протидії шифрувальним вірусам та відновлення зашифрованих файлів шляхом застосування активних дій у комп'ютерному середовищі і утворення каналів впливу на програмні засоби, що реалізують зловмисні криптографічні перетворення. Тому, враховуючи реалії сьогодення, а також брак досвіду роботи з відповідними технологіями ця проблема потребує додаткового і більш глибокого вивчення.

Вирішення задачі забезпечення ефективного захисту доступності інформаційних ресурсів гарантоздатних інформаційних систем (ГІС), які пошкоджені внаслідок несанкціонованих криптографічних перетворень (НКП), потребує побудови адекватної моделі загроз, що в умовах постійного вдосконалення методів та засобів нападу потребує уточнення моделі загроз та моделі порушника [7]. Метою дій порушника є нанесення суттєвих збитків власнику системи, мінімізуючи при цьому власні фінансові, матеріальні та інші витрати. Для цього, можливо вважати, що, по-перше, він має достатньо високу кваліфікацію та необхідний фінансовий ресурс, технічне і програмне

оснащення, які дозволяють йому створювати складні програмно комплекси для реалізації кібератак. По-друге, згідно з принципом Керкхофса [7,8], він знає алгоритми функціонування засобів захисту, включаючи засоби КЗІ, але до початку атаки знає діючих ключів. По-третє, Для досягнення поставлених цілей порушник має можливість перехоплення будь якої інформації що циркулює в інформаційній системі, модифікацію або створення неприпустимої команди за відносно невеликий час.

Виходячи з викладеного, можливо передбачити наступні варіанти його зловмисних дій (потенційних загроз) стосовно ГІС в цілому [15]: 1) модифікація дійсної команди або реальної інформації про внутрішній стан системи; 2) формування та надсилання керованому об'єкту неприпустимої команди або фальшивих даних про внутрішній стан ГІС; 3) перехоплення в транспортній мережі окремих команд або частки інформації щодо внутрішніх станів задля їх вилучення; 4) крадіжка конфіденційної інформації щодо сервісів, які надаються; 5) модифікація або руйнування програмного коду ГІС. 6) шифрування інформаційних ресурсів ГІС за допомогою стійких систем шифрування, включаючи такі, що входять безпосередньо до її складу або операційної системи. Щодо програмних реалізацій засобів КЗІ, які використовуються в ГІС, можливо вважати, що метою дій порушника може бути: 1) зміна, знищення або крадіжка критичних параметрів криптопровайдера CSP (cryptography service provider); 2) модифікація програмного коду (криптосхеми) засобу КЗІ. Водночас, у випадку зловмисного впровадження в небезпечне середовище лише одного з рівнянь (1) практично стійкого криптографічного перетворення при невідомому ключі виключається можливість отримати в системі дані (команди управління, послуги).

2. ФОРМАЛЬНА МОДЕЛЬ КРИПТОГРАФІЧНИХ ПЕРЕТВОРЕНЬ В ГАРАНТОЗДАТНІЙ СИСТЕМІ

Достатньо тривалий час у наукових роботах, що присвячені проблемам інформаційних технологій, використовуються два дуже важливих для побудови сучасних інформаційних систем поняття, а саме, «гарантоздатність системи» [24] і «гарантований захист інформації» [25].

Зауважимо, що існує декілька визначень поняття інформаційної технології (ІТ), зокрема, в [26] ІТ визначена як ресурси, що необхідні для отримання, обробки, зберігання та розповсюдження інформації. В [27] зазначено, що інформаційна технологія у вузькому значенні стосується технологічного боку інформаційної системи і вміщує технічне і програмне забезпечення.

Що стосується гарантоздатності комп'ютерної системи, то спочатку цю характеристику переважно ототожнювали з спроможністю системи забезпечувати в заданих умовах безперебійну роботу і бути відмовостійкою [24]. У подальших наукових публікаціях поняття гарантоздатності було суттєво розширене [27, 28].

Зокрема, ця комплексна характеристика зараз вже включає поняття цілісності, як властивості системи бути незмінною підчас функціонування в умовах випадкових або навмисних руйнуючих впливів, а також конфіденційність, як спроможність системи забезпечувати захист від несанкціонованого використання інформації або технічних засобів [29].

Питання забезпечення гарантоздатності системи в аспекті криптографічного захисту інформації розглянуто в [30] Водночас, у переважній більшості досліджень у галузі технічного захисту ця характеристика системи не фігурує. Постає наступне питання: В яких випадках забезпечення гарантоздатності інформаційної системи потребує впровадження заходів захисту інформації?

ISSN 2663 - 4023

Це питання набуває особливої актуальності у зв'язку з поширенням робіт, які пов'язані з наданням електронних довірчих послуг [31]. У цьому сенсі достатньо згадати визначення гарантоздатності, як комплексної властивості системи надавати послуги, якім *виправдано можливо довіряти* [28], інакше, гарантоздатні системи це високонадійні, відмовостійкі, безпечні і живучі системи з гарантовано достовірними обчисленнями [29].

Слід звернути увагу, що за суттю процесів, які реалізуються, методи криптографії на відміну від методів технічного захисту інформації спрямовані не на обмеження доступу до інформації або блокування каналів її витоку, а саме *на переробку* семантичної (змістовної) інформації або даних управління певним чином, внаслідок чого *інформація набуває нової якості: адресної* (персональної) ***доступності*** її змісту та/або *гарантованого підтвердження джерела* її походження.

Зауважимо, що криптографічна переробка інформації [32], наприклад, у випадку асиметричної криптографічної системи описується двома взаємно пов'язаними математичними рівняннями – прямим та оберненим E, E^{-1}:

$$\begin{cases} E(M, K_e) = C \\ E(C, K_d) = M' \end{cases} \quad (1)$$

де $M, C,\ K_e, K_d, E,\ E^{-1}$ це відповідно: вихідна інформація (дані, команди управління процесом), перетворена інформація (шифровані дані), ключ зашифрування (первинного перетворення), ключ розшифрування (вторинного перетворення), пряме та зворотне перетворення,

Для формального опису криптографічної інформаційної технології у вигляді спрощеної моделі машини Тюрінга, введемо наступні позначення:

$\Rightarrow -$ оператор породження, або передачі, або успадковування об'єкта (процесу) наступним об'єктом (процесом),

$= -$ оператор обчислення значення деякого параметру (математичного об'єкту) або процесу;

$=^{?} -$ оператор перевірки рівності двох математичних виразів, зокрема, перевірки істинності логічного предикату. Якщо умова не виконана, то наступний оператор не виконується (пропускається);

$\in^{?} -$ оператор перевірки належності певного об'єкту (процесу) до деякої множини об'єктів (процесів). Якщо умова не виконана, то наступний оператор не виконується;

$\|$ - оператор конкатенації двох повідомлень (послідовностей даних або команд).

Виходячи з (1) та моделі Шеннона для секретної системи [33] можливо побудувати логічний ланцюг (так званий «цифровий конверт») переробки і передачі інформації від одного суб'єкта (об'єкта) інформаційної системи до іншого:

$$\langle \boldsymbol{A} \rangle \Rightarrow M \Rightarrow E(M, K_e) = C \Rightarrow \langle \boldsymbol{UTE} \rangle \Rightarrow \tilde{C} \Rightarrow E^{-1}\left(\tilde{C}, K_d\right) = \tilde{M} \Rightarrow \tilde{M} \in^{?} \mathfrak{M} \Rightarrow \tilde{M} \Rightarrow \langle \boldsymbol{B} \rangle, \quad (2)$$

де $\langle \boldsymbol{A} \rangle, \langle \boldsymbol{UTE} \rangle, \langle \boldsymbol{B} \rangle -$ відповідно джерело інформації, середовище її поширення та приймач інформації, у якості яких можуть виступати як об'єкти, так і процеси, а $\tilde{C}$, $\tilde{M} -$ відповідно зашифроване повідомлення, що успадковане від середовища $\langle \boldsymbol{UTE} \rangle$, та результат його розшифрування, $\mathfrak{M} -$ множина вихідних повідомлень.

В наведеному логічному ланцюгу завдяки криптографічній переробці інформації вона набуває властивості «не втрачати конфіденційність» підчас її передачі через потенційно небезпечне середовище (*untrusted environment*) – $\langle \boldsymbol{UTE} \rangle$.

При цьому, якщо не виконується $\widetilde{M} \in \mathfrak{M}$ (шифр є імітостійким [11]), то приймач має можливість виправдано відхилити надану інформацію (дані, команду управління), що фактично знижує ризик отримання недостовірної інформації.

Аналогічно будується інший формалізований логічний ланцюг, що забезпечує нову якість інформації – «неможливість її непомітної несанкціонованої зміни» підчас передачі її через потенційно небезпечне середовище або її збереження в ньому:

$$\langle \boldsymbol{A} \rangle \Rightarrow M \Rightarrow E(M, K_{eA}) = M^* \Rightarrow M \parallel M^* \Rightarrow \langle \boldsymbol{UTE} \rangle \Rightarrow E^{-1}(M^*, K_{dA}) =^? M \Rightarrow \langle \boldsymbol{B} \rangle, \quad (3)$$

де K_{eA}, K_{dA} – відповідно секретний та публічний ключі відправника інформації.

Виходячи з побудованих моделей (2, 3) можливо зробити висновок, що криптографічна переробка інформації слугує досягненню головної властивості гарантоздатних систем – забезпечення можливості надання послуг (сервісів), яким виправдано можливо довіряти, навіть у випадку передачі інформації через небезпечне середовище. Водночас, впровадження в небезпечне середовище лише одно з рівнянь (1) у випадку практично стійкого криптографічного перетворення при невідомому ключі виключає можливість

3. МОДЕЛЬ КІБЕРАТАК ІЗ ЗАСТОСУВАННЯМ ЗЛОЯКІСНИХ ШИФРУЮЧИХ КОДІВ

На підставі аналізу наукових публікацій щодо реалізації кібератак з шифруванням інформації [1-3] можливо визначити типову послідовність їх етапів (модель атак) в ГІС (комп'ютерних системах – КС) за допомогою програмних засобів реалізації атак (ПЗРА), яка включає вісім фаз активних дій порушника безпеки (Рис. 1):

Рис. 1 – Етапи реалізації кібератак в ГІС

Можливо наступним чином охарактеризувати кожен з вказаних етапів.

Етап 1 – «Розвідка». На першому етапі порушник, використовуючи всі доступні методи, здійснює приховане вивчення вразливостей комп'ютерної системи (КС), яка є технологічною базою функціонування ГІС, а також виявлення слабких місць наявної системи захисту [2-3].

Етап 2 – «Розробка». На цьому кроці, здійснюється вивчення отриманої інформації та розробка програмних засобів реалізації атак (ПЗРА).

Етап 3 – «Маскування». Порушник здійснює заходи щодо усунення ознак, які пов'язують ПЗРА та спосіб його застосування з реальним розробником, та/або створює фіктивні ознаки, що ототожнюються з непричетними до кібератаки суб'єктами. Також

він визначає тактику приховування реального маршруту (адрес проміжних вузлів глобальної мережі) спроб проникнення в КС.

Етап 4 – «Проникнення». Використовуючи створені засоби і технології, а також можливості інсайдерського впливу порушник забезпечує подолання системи захисту та проникнення ядра ПЗРА в програмне середовище КС.

Етап 5 – «Підготовка». На цьому етапі в автоматичному або автоматизованому режимі реалізується збирання ПЗРА з окремих модулів, його інсталяція та ініціалізація.

Етап 6 – «Реалізація». Зібране та ініціалізоване ПЗРА на основі певної апріорної інформації про підсистеми (елементи) КС, що виконують конкретні функції ГІС, а також про потрібні порушнику дані, зокрема, чутливі параметри безпеки криптографічних модулів SSP, виявляє та ідентифікує зазначені об'єкти у запам'ятовуючих пристроях КС. У якості відповідної апріорної інформації можуть виступати розмір файлів, формати даних, певні ключові слова, програмні переривання/ звернення до деяких ресурсів системи тощо. Залежно від цілей кібератаки виявлені ресурси можуть бути зашифровані, знищені, модифіковані або використані для розкриття конфіденційної інформації.

Етап 7 – «Витік». За необхідності, створюється канал прихованої передачі зібраних даних з використанням методів стеганографії, процедури стискання даних з наступним шифруванням, фізичного переносу під час підключення зовнішніх пристроїв тощо. Наприклад, для створення прихованого каналу передавання даних можливо використовувати 32-х бітове поле *ISN* в *TCP* протоколі [13], яке призначене для забезпечення взаємодії віддаленого клієнта з сервером. Слід зауважити, що досить невеликий обсяг трафіку може бути не виявлений стандартними методами його аналізу.

Етап 8 – «Самоліквідація». На завершальному етапі ПЗРА включає механізми самознищення та приховування слідів кібератаки. Цей етап реалізується автоматично, в разі настання в КС певних обставин (наприклад, визначеного часу), або автоматизовано на підставі отримання команди ззовні. Зокрема, після виконання сумно звісного шифрувального коду Petya ініціюється завдання на перезапуск комп'ютера, що суттєво ускладнює відновлення даних.

Запропонована модель читко відстежується на прикладі шкідливого криптографічного вірусу WannaCry [3]. Цей шкідливий код шукає в мережі комп'ютери з відкритим портом TCP з номером 445, що використовується мережним протоколом прикладного рівня SMBv.1 (*Server Message Block*) для віддаленого доступу до мережних ресурсів та для міжпроцесної взаємодії. У випадку успіху вірус намагається скористатись вразливістю EternalBlue для встановлення «лазівки» DoublePulsar, яка забезпечує завантаження та запуск виконуваного коду WannaCry. WannaCry перевіряє наявність на комп'ютері – мішені «лазівки» DoublePulsar, за допомогою якої завантажується. Математична модель WannaCry її шифрувальної частини може бути описана у вигляді наступної послідовності процедур (PWC1-PWC6).

PWC1. Після запуску WannaCry генерується унікальна для конкретного комп'ютера пара ключів алгоритму RSA: (e, d), де e – відкритий ключ, d – секретний ключ порушника:

$$ed = 1 mod \varphi(N),$$

де $\varphi(N) = (P-1)(Q-1)$, фукція Ейлера,

P, Q – випадкові прості числа, $N = P \cdot Q \geq 2^{2048}$.

Для цього необхідно здійсни декілька звернень до функцій генерації випадкових чисел та тестування чисел на простоту. У загальному випадку вказані функції по

відношенню до можуть бути внутрішніми або зовнішніми, що використовують можливості криптографічних DLL бібліотек операційної системи [6,16].

PWC2. Для чергового файлу комп'ютера F_i (певного типу, зокрема, з розширеннями $.docx, .pdf, jpeg, .pptx$ тощо) генерується унікальний ключ довжиною 128 біт $K_i = (k_{i1}, \dots, k_{i128})$, де $k_{ij} \in \{0,1\}$ для $j = \overline{1,n}$. Виняток становлять файли, які потрібні вірусу для продовження функціонування.

PWC3. Шифрування зазначених файлів відбувається за допомогою симетричного блокового алгоритму AES в режимі CBC:

$$\bar{F}_i = AES_{CBC}(F_i, K_i).$$

PWC4. Кожен ключ симетричного алгоритму K_i шифрується відкритим ключем RSA, результат шифрування $\bar{K}_i$ зберігається в заголовку зашифрованого файлу $\bar{F}_i$:

$$\bar{K}_i \equiv K_i^e mod N, \;\; \bar{F}_i^* = \bar{K}_i \parallel \bar{F}_i.$$

PWC5. Кожен зашифрований файл отримує розширення *.wncry*.

PWC6. Пара ключів RSA ураженої системи шифрується відкритим ключем зловмисника і відправляється на сервери управління, що розташовані в мережі *Tor*, після чого всі ключі з пам'яті інфікованої машини видаляються, а на моніторі з'являється повідомлення з вимогою виплати певної суми у криптовалюті.

Аналіз зазначеної криптографічної схеми свідчить, що у випадку правильного використання криптографічних примітивів, якісної генерації ключів за допомогою фізичних генераторів випадкових чисел, правильного їх знищення на інфікованому комп'ютері відновити зашифровані файли в сучасних умовах майже неможливо.

У той же час, на операційних системах (ОС) Windows XP и Windows Server 2003, якщо комп'ютер не був завантажений наново після його ураження, внаслідок особливостей реалізації ОС алгоритму генерації псевдовипадкових чисел існує можливість відновлення секретних ключів алгоритму RSA і розшифрувати усі перетворені файли. Така можливість досліджена у випадку ОС Windows 7 французькими експертами з компанії Comae Technologies та практично реалізована у вигляді відкритої утиліти WanaKiwi, що надає можливість відновити зашифровані файли [4].

Сучасний етап розвитку криптографічного захисту інформації і стандартизації у галузі криптографії характеризується широкою доступністю так званої «сильної» криптографії, тобто відкритих стандартів практично стійких алгоритмів шифрування, функцій хешування, алгоритмів цифрового підпису, протоколів генерації та управління ключами. Існують вихідні тексти програм, що реалізують відповідні процедури доступні в мережі Інтернет та у спеціалізованих науково-практичних виданнях [4,16]. Відповідно, складність задач атакуючої сторони щодо створення криптографічних ПЗРА суттєво спрощується, а ефективність вирішення задач по відновленню (дешифруванню) даних КС, які зашифровані за допомогою ПЗРА, має об'єктивну тенденцію до постійного з року в рік зниження.

У зв'язку з цим, особливий інтерес становлять сучасні технології, що дають можливість підвищити ефективність класичних методів дешифрування [7-11], які використовуючи вразливості в певному криптографічному алгоритмі та/або протоколі забезпечують відновлення секретного ключу, за допомогою якого створені зашифровані повідомлення, та/або розкриття вихідного значення зашифрованої

ISSN 2663 - 4023

інформації. При цьому відновлення вихідної інформації може бути повним або частковим, тому говорять про повне або часткове дешифрування.

4. ХАРАКТЕРИСТИКА ОБЧИСЛЮВАЛЬНОЇ СКЛАДНОСТІ ЗАДАЧ ВІДНОВЛЕННЯ ЗАШИФРОВАНИХ ДАНИХ

Зважаючи на швидке зростання потужності сучасних КС, можливість використання ресурсів різного роду аутсорсінгових центрів, виникає ідея застосування цих потужностей для відновлення зашифрованих даних, тобто вирішення задачі криптоаналізу за методом «грубої сили».

Практичну складність розв'язку задач криптоаналізу звичайно оцінюють кількістю елементарних операцій W, що необхідні для повної реалізації деякого найкращого з усіх відомих алгоритму. З точки зору забезпечення надійного захисту інформації, складність вважається достатньою, якщо для найкращого алгоритму дешифрування час розв'язку задачі за допомогою найпотужнішої обчислювальної техніки суттєво перевищує "час життя" конфіденційної інформації T_I. Вказаний час розуміється при цьому як термін, після спливу якого втрачається сенс зберігати інформацію у таємниці (у випадку протидії ПЗРА – відновлювати інформацію).

Для алгоритму DES (*Data Encryption Standard*), складність розв'язку задачі знаходження, 56-бітного ключа оцінюється величиною $W = 2^{56} \approx 10^{17}$ (методи диференційного та лінійного аналізу на практиці не можуть бути застосовані [4]). Таким чином інформація, яка зашифрована за допомогою DES, може бути відновлена за методом «грубої сили» вже при потужності обчислювальної системи, що здатна повіряти $w = W/T_I \approx 10^8$ ключів за секунду.

З точки зору криптоаналітика, продуктивність обчислювальної системи достатня, якщо за наявності у його розпорядженні найкращого алгоритму, складність реалізації якого $W_{еф}$, час знаходження ключу або відкритого повідомлення не перевищує деякої оперативне прийнятної величини $T_{оп}$, після чого подальший розв'язок задачі не має сенсу. При цьому необхідна продуктивність обчислювальної системи оцінюється як:

$$w = W_{еф} / T_{оп} .$$

Оскільки оперативний час залежно від умов обстановки може змінюватися від декількох хвилин до декількох місяців, то і потужність обчислювальної системи для цілей криптоаналізу повинна бути відповідною.

В ряді випадків скоротити загальний час розв'язку задач криптоаналізу можливо за рахунок її розбивання на n підзадач, які опрацьовуються одночасно, наприклад, шляхом розпаралелювання процесів. Це, відповідно, може привести до зменшення часу розв'язку приблизно в ті ж самі n разів.

Нині можливо виділити такі методи розпаралелювання розв'язку задач криптоаналізу:

подання вихідної множини невідомих параметрів (ключів) у вигляді підмножин, що не перетинаються, після чого пошук рішення здійснюється одночасно (синхронно) з наступним корегуванням результату на парах підмножин за необхідності. У разі знаходження рішення, що задовольняє встановленим критеріям для будь якої підмножини параметрів, обчислювальний процес зупиняється;

ISSN 2663 - 4023

виділення безпосередньо в алгоритмі розв'язку задачі обчислювальних процедур, які можуть виконуватися одночасно (паралельно);

поділ задачі на аналітичну (алгебраїчну) та перебірну частини, що можуть виконуватися синхронно;

комбінування вказаних методів.

Основні методи, що використовуються для розв'язку задач оцінювання стійкості криптосистем, пошуку вразливості та дешифрування залежно від базових математичних методів можливо умовно розділити на декілька класів, а саме:

імовірнісно-статистичні методи (ІСМ) дослідження великих обсягів (до $N \approx 2^{100}$) вихідних послідовностей псевдовипадкових генераторів ключових даних або послідовностей, які зашифровані за допомогою засобів КЗІ;

методи ортогональних перетворень (МОП) вихідних аналогових та цифрових даних за методами Фур'є, Уолша, Уолша-Адамара, вейвлет - перетворень, обчислення функцій згортки та кореляції;

алгебраїчні та комбінаторні методи розв'язку (АКМР) систем лінійних або нелінійних рівнянь великої розмірності ($N \approx 2^{80} \div 2^{256}$) з викривленими правими частинами, розкладання великих чисел на прості множники ($N \approx 2^{1024} \div 2^{2048}$), знаходження дискретного логарифма в полі великого порядку ($N \approx 2^{1024} \div 2^{2048}$), знаходження кратності точки на еліптичної кривої тощо;

методи лінійного і динамічного програмування (МЛДП) розв'язку задач пошуку екстремумів функцій, заданих на просторі ключів (розкриття ключів типу підстановка або перестановка);

методи оптимізованого або повного перебору (МОПП) параметрів криптосистем;

змішані методи.

Серед ІСМ методів особливим значенням для криптоаналізу та найбільшою обчислювальною складністю характеризуються методи пошуку пар послідовностей за критерієм "нульових вертикальних біграм" (НВБ), методи пошуку суцільних та переривчастих повторів (ПСПП) за відповідним критерієм, а також різного роду комбінації цих методів. За їх допомогою можливо виявити випадки однаково зашифрованих повідомлень.

Складність зазначених методів можна оцінити такими величинами:

$$W_{НВБ} \approx K_1 \cdot C_N^2 \cdot 2 \cdot (M - m), \tag{1}$$

$$W_{ПСПП} \approx K_2 \cdot \left[\frac{M \cdot N}{L}\right] \cdot \lg\left[\frac{M \cdot N}{L}\right], \tag{2}$$

де N – кількість послідовностей, що аналізуються;

M – середня довжина послідовності;

L – довжина повторення;

m – мінімальна довжина "перетинання" двох послідовностей для отриманні сталого статистичного виводу (нормальної роботи критерію);

K_1, K_2 - постійні коефіцієнти, що залежать від системи команд процесору та використаної мови програмування.

З формул (1) та (2) можливо отримати практичні оцінки для складності розв'язку відповідних задач. Наприклад, якщо кількість послідовностей, що аналізуються дорівнює $N = 10^5$, їх середня довжина становить $M = 10^3$ біт, а мінімальна довжина "перетинання" - $m = 10^2$, то складність розв'язку задачі підбору пар за критерієм НВБ оцінюється

величиною $W_{НВБ} \approx 9 \cdot 10^{12}$. Це означає, що за наявності обчислювальної потужності системи w понад 10^8 елементарних операцій в секунду, задача комплектування може бути вирішена протягом доби. Якщо ж у попередніх умовах довжина повторення L дорівнюватиме 10, то нескладно бачити, що за наявності обчислювальної потужності $w \geq 7 \cdot 10^7$ операцій за секунду повтори довжиною в 10 біт можливо виявити за одну добу. Наведені оцінки мають значення для викладення змісту третьої частини роботи та побудови атак на стійкі криптографічні системи із застосуванням технологій активного впливу у кібернетичному просторі на криптографічні примітиви, зокрема, генератори випадкових даних.

Зауважимо, що пошук повторів можливо здійснити значно швидше, якщо обчислювальна система має швидкісну пам'ять з прямим доступом відповідного об'єму. Наприклад, в умовах наведеного прикладу для фіксації усіх повторів довжиною 10 біт необхідно мати 2^{10} комірок пам'яті, у яких фіксують координати (номер послідовності, номер місця з якого починається повтор) кожного можливого повтору. При цьому розмір комірки безпосередньо залежить від об'єму вихідного масиву та від ступеню нерівномірності розподілу зустрічаємості елементів послідовностей (в одному крайньому разі комірка може не містити жодного елемента, в другому – усі можливі координати десятиграм).

Використання МОПП характерно для багатьох задач радіотехніки, тому побудова відповідних спеціалізованих обчислювальних систем в інтересах криптоаналізу полегшується за рахунок застосування різного роду готових апаратно-програмних прискорювачів. Проблеми їх застосування мають чисто технічний характер, обумовлений сумісністю різних платформ і форматів подання даних. Зокрема, під час дешифрування сигналів захищеної аналогової телефонії використовується швидке перетворення Фур'є для побудови різного роду цифрових фільтрів. Відомо, що фільтрація сигналів складними не рекурсивними фільтрами, які містять п'ятдесят і більш членів у математичних виразах, що їх описують та велике число повторних обчислень [17].

Якщо вхідний сигнал має M відліків, а нерекурсивний фільтр має N відгалужень (де $M > N$), для оцінки частотної характеристики фільтра потрібно $N \cdot M$ множень. Т.Стокхем [17] запропонував метод зменшення числа множень, що необхідні для оцінювання частотної характеристики фільтра. Він показав, що у випадку N у формі 2^n при правильно організованих обчисленнях число множень приблизно оцінюється такою величиною:

$$W \approx M \cdot \log_2 N. \qquad (3)$$

АКМР відіграли суттєву роль у розробці методів дешифрування потокових криптосистем покоління 60-80 років минулого сторіччя, але питання їх використання для цілей дешифрування сучасних криптосистем блокового типу залишається відкритим. Складність розв'язку систем лінійних рівнянь над двійковим полем F_2 класичним методом Гауса становить $W = n^3/3$, де n - число змінних [8]. У випадку системи лінійних рівнянь над полем із q елементів F_q складність розв'язку за методом І.В. Коновальцева оцінюється величиною:

$$W \approx K \cdot \frac{n^3}{\log_q N}. \qquad (4)$$

Застосування АКМР дає можливість обмежити вимоги до потужності обчислювальної системи, але їх застосування можливе лише за умов знаходження ефективних способів лінеаризації функцій перетворень для конкретних сучасних алгоритмів.

МЛДП досліджені достатньо глибоко. Їх найбільше застосування припало на 50-80 роки минулого сторіччя. На жаль, для сучасних криптосистем їх застосування, в силу ряду причин, залишається проблематичним.

Застосування МОПП параметрів криптосистем, не зважаючи на досить велику довжину ключів сучасних криптосистем залишається одним з перспективних напрямів дешифрування, оскільки у цьому випадку, вочевидь, просто вирішується подання ключового простору у вигляді об'єднання множин меншої потужності. Можливість розв'язання задач дешифрування при цьому визначається технологією побудови та архітектурою швидкодіючої спеціалізованої обчислювальної системи.

5. ЗАСТОСУВАННЯ ТЕХНОЛОГІЇ РОЗПОДІЛЕНИХ ОБЧИСЛЕНЬ ДЛЯ ВИРІШЕННЯ ЗАДАЧ ВІДНОВЛЕННЯ ЗАШИФРОВАНИХ РЕСУРСІВ

Очевидним шляхом розв'язання проблеми відновлення зашифрованих ресурсів є застосування методу «грубої сили» для пошуку істинних значень застосованих ключів. Нескладно показати, що якщо кількість різних криптографічних ключів у симетричній криптосистемі дорівнює N, а максимальний час для відновлення інформації не повинен перевищувати T, тоді потужність обчислювача має бути що найменш $V \geq {}^{N}/_{2T}$ перевірених ключів в одиницю часу.

Висока продуктивність сучасної обчислювальної техніки досягається, як відомо, за рахунок :

- застосування певної ефективної архітектури обчислювальної системи; підвищення тактової частоти роботи інтегральних схем;
- збільшення об'єму вбудованої в мікропроцесори пам'яті;
- збільшення інформаційної ємності та скорочення часу доступу до оперативних запам'ятовуючих пристроїв;
- підвищення швидкодії інтерфейсних шин;
- збільшення кількості мікропроцесорів (ядер) на одній обчислювальній платформі тощо.

Практичну ефективність у випадку розв'язку задач криптоаналізу довели кластерні системи. Розподілені обчислення є окремим випадком технології паралельних обчислень, коли одночасний розв'язок різних частин однієї обчислювальної задачі реалізується декількома процесорами (або ядрами одного процесора) одного або декількох комп'ютерів. При цьому вказана обчислювальна задача сегментується, тобто поділяється на підзадачі, які можуть опрацьовуватися одночасно. Розпаралелювання операцій при реалізації методів криптоаналізу, що наведені вище, часто суттєво залежить від архітектури конкретної розподіленої обчислювальної системи. Найбільш відомими у цьому напрямі є факти реалізації розподілених операцій шифрування/розшифрування, що реалізовані, наприклад. у деяких типах високо швидкісних *IP*-шифраторах. Зауважимо, що таке технічне рішення, у разі доповнення блоком критерію перевірки правильності дешифрування можливо використовувати для побудови спеціалізованої системи пошуку ключів з використанням МОПП. МОПП загалом, як було зазначено раніше, є найбільш простим та зручним інструментарієм у класі паралельних обчислень, що може бути ефективно

використано для практичного дешифрування симетричних і асиметричних криптосистем, пошуку колізій функцій хешування з метою "зламу" парольної автентифікації та криптографічних протоколів, які використовують електронний цифровий підпис. Технічна реалізація розподіленої обчислювальної системи передбачає застосування певної апаратної платформи та розробку мережного програмного забезпечення, що сприятиме об'єднанню окремих обчислювачів у єдину багатопроцесорну систему (далі – БС).

Серед кількох типів окремих обчислювачів, які вельми ефективно об'єднуються в БС, можливо навести приклад станцій з багатоядерними процесорами типу *SUN UltraSPARC T1/T2*.

Кожне з восьми ядер у процесорі підтримує чотири (у $T1$) або вісім (у $T2$) апаратних потоків (*thread*), що звичайним множенням дає 32 або 64 потоки у процесорі. Теоретично це означає, що такий процесор може забезпечити виконання 32/64 задачі однопотокового процесору за той же час. Крім того, слід звернути увагу на технічні можливості графічних юнітів (*GPU*), наприклад, *nVidia GeForce 9600 GT*, який має 64 ядра .

Достатньо часто, для об'єднання *GPU* в БС використовується світч *Foundry SuperX* та гигабітна мережа з ОС *Linux Rocks*. При цьому самим коштовним елементом системи є саме світч.

Інший варіант організації паралельного обчислювача – кластер. Це комплекс взаємозв'язаних та узгоджено функціонуючих апаратних і програмних компонент, який відноситься до класу багатопроцесорних систем, а саме до паралельних обчислювачів. Їх функціонування організоване за принципом *Multiple Instruction – Multiple data* (*MIMD*), що перекладається як "множинний потік команд та множинний потік даних". Як правило, кластер складається з потужного комп'ютеру, комунікаційного обладнання, операційних систем та мережних застосувань.

Багатопроцесорні системи, що об'єднані у кластер, можуть бути охарактеризовані таким чином. Вони мають:

можливість одночасно та незалежно один від другого виконувати декілька програмних гілок;

розділяєму (спільну) або розподілену (індивідуальну) пам'ять, що забезпечується вибором відповідної адресної системи;

масштабовану архітектуру, тобто не встановлюють жорстких обмежень щодо апаратної платформи обчислювальних модулів, щодо обладнання і топології обчислювальної мережі, щодо конфігурації та діапазону потужності обчислювальних засобів.

Найпростіші кластерні комплекси можна побудувати на базі локальних обчислювальних мереж, а спеціалізовані доцільно створювати на основі симетричних мультіпроцесорів (*SMP – Symmetric Multi-Processors*), які, в свою чергу, складаються з декількох однорідних процесорів і масиву спільної пам'яті.

Ще одним шляхом підвищення потужності спеціалізованих обчислювальних комплексів є створення криптографічних процесорів, які реалізують необхідні криптографічні перетворення за мінімальний час. Для цього є серйозне підґрунтя у вигляді програмованих логічних інтегральних схем (ПЛІС). Наявність останніх саме й створює передумови для перегляду підходів до побудови архітектури обчислювальних систем, які призначені для розв'язку задач криптоаналізу. Останнім часом провідні світові виробники (*Altera, Xilinx, Actel, Atmel, Motorola*) створили багатий спектр ПЛІС,

ISSN 2663 - 4023

що мають відносно невелику ринкову вартість, надвелику еквівалентну вентильну ємність та високі електричні параметри. Поява ж на світовому ринку пристроїв зі змінною архітектурою (від компаній *picoChip, QuickSilver, SGI*) взагалі дає можливість однозначно стверджувати, що означена тенденція розвитку мікроелектронної техніки може забезпечити технічні характеристики таких засобів ще на порядок вищими по продуктивності та з набагато меншим рівнем енергоспоживанням.

Сучасні ПЛІС є універсальною елементною базою, що конфігурується залежно від конкретних вимог прикладних застосувань. Універсальність цих схем забезпечується певною топологію, практично єдиною концепцією побудови для будь яких виробників та назв складових топологічних структур. Мікросхеми програмованої логіки будуються за ієрархічним принципом – від простого блоку до складної структури. На першому ієрархічному рівні сучасних ПЛІС бачимо блок, що містить 4-х входову таблицю відповідності, мультіплексор, тригер та спеціальну логіку швидкого переносу. На другому рівні існує блок, що містить декілька блоків 1-го рівня та спеціальну логіку швидкого переносу, и на третьому – блок, що включає декілька блоків 2-го рівня, спеціальну логіку швидкого переносу та відокремлену маршрутизуючу мережу (3-й рівень ієрархії в ПЛІС фірми XILINX називається конфігуруємими логічними блоками, в фірмі ALTERA – блоками логічних масивів).

При реалізації даної ієрархії фірми-розробники залежно від паспортних технічних характеристик ПЛІС, обирають певну варіацію числам елементів на 1-му рівні та рівневі вкладення у цілому. Логічна підтримка визначеної архітектури забезпечується додатковою еквівалентною ієрархією внутрішніх з’єднань. Ієрархічно внутрішні з’єднання поділяються на швидкі на 1-му рівні, на 2-му рівні - менш швидкісні та ще менш швидкісні на 3-му рівні. Таким чином, об’єднання від десятків до ста тисяч і більш блоків 3-го рівня, реалізує архітектуру, топологія якої настроюється. Ця модель фактично являє собою багатошарову штучну нейронну мережу.

Підсумовуючі слід зазначити, що реалізація на базі ПЛІС алгоритмічно завершеного рішення (так званого інформаційного графа алгоритму) з можливістю її оперативного реконфігурування, як складової реалізації деякого загального алгоритму функціонування системи, дозволяє розглядати цей напрям у ДРР як стратегічну можливість створення високо потужного адаптивного обчислювального комплексу для криптоаналітичних застосувань.

6. АТАКИ НА ПРОГРАМНУ РЕАЛІЗАЦІЮ СТІЙКИХ КРИПТОГРАФІЧНИХ АЛГОРИТМІВ

Проблема відновлення інформації у ГІС за оперативно приємний час постійно загострюється. За результатами аналізу сучасних наукових публікацій у відкритих виданнях можливо зробити висновок, що ефективність крипто аналітичних атак на переважну більшість алгоритмів, що використовуються для криптографічного захисту інформації в ГІС, включаючи стандарти *AES, NESSIE*, ГОСТ 28147-89 та інших [18-21], вкрай низька. Але ситуація навколо стійкості криптографічних ПЗРА є не зовсім “райдужною” для авторів програм-вимагачів або зовсім безнадійною для захисту

інформації, якщо криптографічну систему розглядати як сукупність алгоритмів криптографічного перетворення інформації, механізмів генерації ключів та допоміжних параметрів, криптографічних протоколів, технічної або програмної реалізації вказаних математичних функцій. При цьому саме ***програмна*** реалізація може бути вразлива до активних дій у кібернетичному просторі, внаслідок чого вона може бути виявлена в ГІС на основі її ознак та зазнавати модифікації. У цьому випадку для побудови ефективних атак можливо використовувати прорахунки в реалізації криптографічних ПЗРА, що не були виявлені під час його проектування [15, 22].

При реалізації криптографічних алгоритмів в рамках, наприклад, криптопровайдерів ОС *Windows*, а також при використанні криптопровайдерів із застосувань можливі наступні групи загроз, що пов'язані з надійністю [15, 16, 22]:

- Помилки програмування криптографічних примітивів;
- Помилки використання криптографічних примітивів у криптопровайдерах;
- Помилки передачі параметрів у криптопровайдер і поверненні результатів обробки;
- Помилки її відмови апаратної платформи;
- Випадкові та навмисні порушення цілісності програм і даних крипто провайдерів.

Ці фактори можуть бути є передумовами утворення прихованих каналів витоку інформації про роботу криптографічних ПЗРА.

Завдання системи захисту ГІС – створення та впровадження програми - "кібернетичного робота", що здатний контролювати певні події у системі, модифікувати відповідним чином програмні коди ПЗРА, що виконуються, здійснювати підміну деякої інформації.

Розглянемо два варіанти проведення атак на реалізацію ПЗРА, що не передбачають наявності критичної інформації про їх роботу.

Варіант 1. Існує певний криптографічний ПЗРА (криптосистема) з секретними ключами на основі стійкого блокового симетричного криптографічного алгоритму, який забезпечує шифрування у режимі *OFB* (рис. 3) або інакше - гамуваня зі зворотнім зв'язком [10]. Необхідно: побудувати атаку на реалізацію вказаної криптосистеми, щоб забезпечити можливість часткового або повного відновлення інформації (дешифрування).

Такий режим роботи криптоалгоритму досить широко використовується для побудови повно зв'язаних мереж зв'язку, у яких кожен абонент має надіслати повідомлення будь якому іншому абоненту цієї мережі.

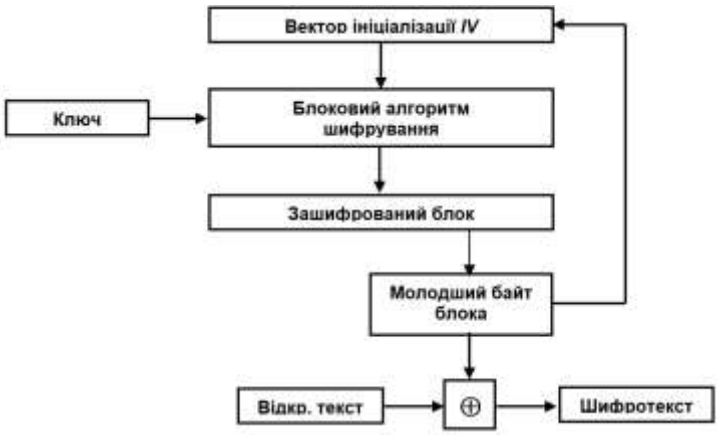

Рис. 3. - Застосування блокового симетричного алгоритму в режимі OFB

ISSN 2663 - 4023

Для цього режиму характерним є наявність вектору ініціалізації IV - випадкового числа, що забезпечує на припустимому рівні ймовірність не перекриття шифру у разі використання одного ключу K протягом певного часу [10]. Ідея атаки на цю криптосистему полягає у реалізації фіктивного генератору випадкових даних криптоалгоритму, що включається замість реального PRNG (pseudorandom number generator) забезпечує приховане повторення векторів ініціалізації або ключів. Внаслідок цього деякі повідомлення будуть зашифровані однаково:

$$\begin{cases} \tilde{S}_1 = \tilde{T}_1 + \tilde{\Gamma}_j \\ \dots \quad \dots \quad \dots \\ \tilde{S}_{m_j} = \tilde{T}_{m_j} + \tilde{\Gamma}_j \end{cases},$$

де $\tilde{T}_1$, ..., $\tilde{T}_{m_j}$ – відкриті повідомлення;

$\tilde{\Gamma}_j$, $j = \overline{0.2^k - 1}$ – послідовність знаків гами, що утворені з одного вектору ініціалізації та ключу алгоритму;

$\tilde{S}_1$, ..., $\tilde{S}_{m_j}$ – відповідні зашифровані повідомлення.

У цьому випадку, залежно від кількості однаково зашифрованих повідомлень m_j та надлишковості вихідних текстів повідомлень відповідні зашифровані повідомлення можуть бути частково, або повністю дешифровані [8]. Для повторення векторів ініціалізації IV генератор випадкових біт доцільно використовувати таким чином:

$$IV = \langle \varphi(\alpha_1, \alpha_2, ..., \alpha_k) \rangle = \langle \beta_1, \beta_2, ..., \beta_b \rangle,$$

де $\alpha_1, \alpha_2, ..., \alpha_k$ - послідовність випадкових біт,

$\varphi(...)$ - деяка однозначна функція, що забезпечує розширення випадкової послідовності до заданого розміру b;

$\beta_1, \beta_2, ..., \beta_b$ - координати вектору ініціалізації.

З метою рішення задачі спочатку розрахуємо припустиму кількість випадкових біт для забезпечення необхідної кількості повторів однаково зашифрованих повідомлень. Нехай у мережі присутні M абонентів, кожен з яких у середньому щодоби відправляє $\bar{\mu}$ повідомлень за кожним напрямом з'єднань. При цьому T - термін дії спільного ключу (діб). Тоді середня кількість відправлених повідомлень $\bar{N}$ становить величину:

$$\bar{N} = \binom{M}{2} \cdot \bar{\mu} \cdot T.$$

Протягом вказаного періоду середня кількість повторів $\bar{R}$ кожного із 2^k значень векторів випадкових біт $\langle \alpha_1, \alpha_2, ..., \alpha_k \rangle$ у випадку їх рівномірного розподілу становитиме:

ISSN 2663 - 4023

$$\overline{R} = \frac{\overline{N}}{2^k} = \binom{M}{2} \cdot \overline{\mu} \cdot T \Big/ 2^k \quad . \tag{5}$$

Виходячи з рівняння (5) отримуємо оцінку припустимого числа випадкових біт:

$$k \le \left\lfloor \log_2 \binom{M}{2} \cdot \overline{\mu} \cdot T - \log_2 \overline{R} \right\rfloor . \tag{6}$$

За допомогою нерівності (6) у кожному конкретному випадку мережі, що атакується, можливо розрахувати припустиму кількість випадкових біт, для забезпечення середньо статичної кількості $\overline{R}$ однаково зашифрованих повідомлень. Звичайно, випадкові дані, що використовуються для криптографічних перетворень підлягають статистичному тестуванню, але досить мала довжина вектору IV (довжина блоку для більшості стандартних криптоалгоритмів дорівнює *64* або *128* біт) суттєво обмежує можливості щодо застосування статистичних критеріїв. Для перевірки рівномірності розподілу послідовностей такої довжини переважно застосовуються критерії частот знаків та біграм [23], що висуває відповідні вимоги до функції розширення. Тому зрозуміло, що для забезпечення рівної зустрічаємості біграм у двійковому векторі $\langle \alpha_1, \alpha_2, ..., \alpha_k \rangle$ доцільно величину k обрати кратною *4* (всього існує чотири варіанти біграм 00, 01, 10, 00).

Наприклад, у разі мережі спеціального зв'язку що включає $M = 10$ абонентів із середньою інтенсивністю відправлення повідомлень протягом терміну дії одного мережного ключу $\overline{\mu} \cdot T = 10$ для отримання середньої кількості повторів шифрування $\overline{R} = 2$ необхідно мати кількість випадкових біт не більш $k \le 7$.

Для приховування статистичних залежностей в векторі *IV* та отримання необхідної кількості біт у відповідному виразі у якості функції φ слід скористатися безумовно стійкою хеш–функцією, наприклад *MD*5, яка входить до стандартного набору алгоритмів ОС *Windows,* або аналогічною іншою. При цьому зі 128 біт дайджесту, що отримуємо за допомогою хеш-функції, скористаємося необхідною кількістю.

Варіант 2. Нехай спостерігається ПЗРА (криптосистема), що побудований на основі стійкого симетричного блокового криптографічного алгоритму шифрування у режимі *OFB* з використанням відкритого розподілу ключів на основі протоколу Ель-Гамаля [10].

Задача полягає у організації атаки на реалізацію криптосистеми таким чином, щоб забезпечити повне дешифрування.

Спочатку роботи криптосистеми у протоколі Ель-Гамаля за допомогою циклічного елемента g деякого поля та секретного ключу асиметричного алгоритму x формується відкритий ключ y:

$$y = g^x \bmod p$$

де p – деяке велике просте число, довжиною від 1024 біт.

Для зашифрування сеансового ключу симетричного алгоритму генерується випадкове число k, обчислюється величина:

$$y_1 = g^k \bmod p$$

та шифрується секретний ключ симетричного алгоритму K:

$$\delta = K \cdot y^k \bmod p \quad (7)$$

Пара (δ, y_1) разом із зашифрованим повідомленням передається власнику секретного ключу x, який на основі отриманої пари (δ, y_1) обчислює секретний ключ симетричного алгоритму K:

$$K = \delta \cdot y_1^{-x} \bmod p$$

Атаку на вказаний протокол організуємо шляхом підміни випадкового числа k псевдовипадковим таким чином, щоб система тестування не виявила цього факту. Нехай $k \in \{\xi_t, t=1,2, \dots, \Psi\}$, при цьому потужність Ψ множини припустимих значень псевдовипадкового числа k виберемо досить великою, але достатньо меншою ніж продуктивність спеціалізованої обчислювальної системи щодо перебору ключів за припустимий проміжок часу. Обчислення можливих варіантів ключа K у цьому випадку здійснімо за допомогою МОПП виходячи з рівняння (7) на підставі множини припустимих значень псевдовипадкового числа $\{\xi_t\}$. У підсумку слід зауважити, що доступ к інформації «істинного» генератору випадкових даних та методи його підміни у кожному конкретному випадку залежать від апаратної та програмної платформ, на яких функціонує ПЗРА, особливостей його реалізації, наявності в складі автоматизованої системи засобів захисту від несанкціонованого втручання в її роботу тощо. Для створення фіктивного генератору випадкових простих чисел можливо скористатись послідовністю чисел Мерсенна $M_p = 2^p - 1$, якщо p – просте число, M_p також просте.

З урахуванням викладеного уявляється сформувати наступну семантичну модель протиборства захисту ГІС та атакуючої сторони.

Таблиця 1

Семантична модель протиборства захисту ГІС та атакуючої сторони

Дія	Час початку	Етап НКП	Методика дій захисту ГІС
A_1	t_1	Вибір об’єкту атаки	Апріорна оцінка загрози
A_2	t_2	Пошук вразливості	Моніторинг портів та підключення пристроїв
A_3	t_3	Вбудовування Backdoor (лазівки), модифікація коду системи	Контроль несанкціонованих дій
A_4	t_4	Завантаження та запуск шкідливого коду	Контроль НСД, обчислення функції прийняття рішення
A_5	t_5	Активація асиметричного перетворення, виклик стандартних функцій PRNG, CryptoAPI тощо	Контроль, перехоплення звернень, підміна фіктивною функцією
A_6	t_6	Активація симетричного перетворення, виклик стандартних функцій PRNG, CryptoAPI тощо	Контроль, перехоплення звернень, підміна фіктивною функцією
A_7	t_7	Читання файлів за переліком форматів	Контроль і управління доступом

		(розширень)	
A_8	t_8	Заміна розширення зашифрованого файлу на нестандартний формат	Контроль та блокування процесу
A_9	t_9	Знищення ключових даних шляхом звернення к відповідним функціям	Контроль та блокування процесу
A_{10}	t_{10}	Відправлення зашифрованих даних зловмиснику	Контроль та блокування процесу
A_{11}	t_{11}	Перезавантаження системи	Контроль та блокування процесу
		Закінчення атаки	Пошук ключів у ОЗП, RAM

Для семантичної моделі часові інтервали мають наступні обмеження:

$$\begin{cases} t_{11} - t_1 \le T_{\text{атаки},max} \\ t_{i+1} - t_i \le \Delta_i \end{cases}, \quad (8)$$

де $T_{\text{атаки},max}$ – загальна тривалість атаки, Δ_i – середній час між етапами атаки.

В умовах (8) на кожному етапі $\{A_i, i = \overline{1,11}\}$ ми розраховуємо функції прийняття рішення про атаку:

$$\vartheta_i(A_1, \quad \dots, \quad A_i) = \begin{cases} 1, \text{якщо приймається рішення про атаку} \\ 0, \text{ в іншому випадку} \end{cases}$$

Результати проведених комп'ютерних експериментів свідчать, про достатню ефективність запропонованої методики у частині протидії НКП та відновлення зашифрованої інформації та можливість її широкого застосування в системах антивірусного захисту.

7. ВИСНОВОК

Підсумовуючі викладене можливо зробити такі висновки.

1) В сучасних умовах постійного розвитку методів криптографії з високім рівнем стійкості та їх широкою доступністю необхідною умовою підвищення ефективності відновлення зашифрованих програмами – вимагачами даних є створення спеціального програмного монітору безпеки та побудови спеціалізованих багатопроцесорних систем для реалізації методів криптоаналізу із широким доступом авторизованих користувачів.

2) З точки зору мінімізація співвідношення "вартість - отриманий результат" найбільш раціональним підходом до створення спеціалізованих багатопроцесорних систем є побудова кластерної системи на базі найбільш потужних комп'ютерів загального призначення із застосуванням апаратних прискорювачів обчислень на базі ПЛІС.

3) Для підвищення ефективності атак на криптографічні ПЗРА доцільно розвивати технології активних дій у кібернетичному просторі, зокрема, такі, що забезпечують утворення прихованих каналів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

[1] A. Young, M. Yung Cryptovirology: extortion-based security threats and countermeasures, Published in: Proceedings 1996 IEEE Symposium on Security and Privacy, 6-8 May 1996, Oakland, CA, USA, Publisher: IEEE DOI: 10.1109/SECPRI.1996.502676 ISBN: 0-8186-7417-2, Pp. 125-140

[2] A. Young, M. Yung. Malicious cryptography exposing cryptovirology. Wiley Publishing, Inc., p. 392, 2004.

[3] Clark., Z., 2017. The Worm That Spreads Wanacrypt0r. [online] Malwarebytes. Available at: <https://blog.malwarebytes.com/threat-analysis/2017/05/the-worm-that-spreads-wanacrypt0r/> [Accessed 5 October 2020].

[4] Suiche M., 2017. WannaCry Decrypting files with WanaKiwi + Demos. [online] Comae Technologies. Available at: https://blog.comae.io/wannacry-decrypting-files-with-wanakiwi-demo-86bafb81112d. [Accessed 5 October 2020].
[5] S.A. Shivale, "Cryptovirology: Virus Approach ", *International Journal of Network Security & Its Applications (IJNSA)*, № 3(4), p.33-46, 2011.
[6] Kaur, J., Jaafar F. and Zavarsky P., 2018. An Empirical Analysis of Crypto-Ransomware Behavior. In: The Thirteenth International Conference on Systems (ICONS 2018). pp.1-7.
[7] Горбенко І.Д., Горбенко Ю.І. Прикладна криптологія: Теорія. Практика. Застосування. Монографія. – Харків, ФОРТ, 2012, 880 с.
[8] Бабаш А.В., Шанкин Г.П., Криптография, М.: СОЛОН-Р, 2002, -512с.
[9] Bauer F. Decrypted Secrets: methods and maxims of cryptology. N.Y.: Springer, p. 472, 1997.
[10] Шнайер Б. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си. – М.: Триумф, 2002. – 816 с.
[11] Alan G. Konheim, Computer Security and Cryptography, Published by John Wiley & Sons, Inc., Hoboken, New Jersey, p. 521, 2007.
[12] Грушко, А., 1999. О существовании скрытых каналов. Дискретная математика, 1(11), pp.24-28.
[13] Rowland C. H. Covert channels in the TCP/IP protocol suite. 1997, First Monday, 2(5). https://doi.org/10.5210/fm.v2i5.528
[14] J. Kelsey Side Channel Cryptanalysis of Product Ciphers / J. Kelsey, B. Schneier, D. Wagner, C. Hall // 5th European Symposium on Research in Computer Security Louvain-la-Neuve, Belgium September 16–18, 1998 Proceedings, Berlin, Springer, 1998, pp.97-111.
[15] Щербаков А.Ю., Домашев А.В., Прикладная криптография. Использование и синтез криптографических интерфейсов. – М.: Издательско-торговый дом «Русская редакция», 2003, 416 с.: ил.
[16] Гулак Г.М., Забезпечення безпеки засобів КЗІ у кіберпросторі. Матеріали науково-технічної конференції «Сучасні інформаційно-телекомунікаційні технології», том ІУ Сучасні технології інформаційної безпеки, -К.:-2015, с.100-102
[17] Голд Б., Рейдер Ч. Цифровая обработка сигналов. – М.,: «Сов. Радио», 1973, с. 368.
[18] Горбенко И., Гулак Г., Олейников Р., Шумов А., Горбенко Ю. 2003, Основные принципы проектирования оценка стойкости и перспективы использования в Украине алгоритма шифрования AES //Научно-технический сборник "Правовое, нормативное и метрологическое обеспечение", (7), с.14
[19] Горбенко І.Д., Гулак Г.М., Олійников Р.В., Руженцев В.І., Михайленко М.С., 2005, Аналіз властивостей алгоритмів блокового симетричного шифрування (за результатами міжнародного проекту NESSIE) //8 Международная научно-практическая конференция «Безопасность информации в информационно-телекоммуникационных системах». Тезисы докладов. –К. 2005. с. 17-18
[20] Рудской, В., О нулевой практической значимости «Атаки определения ключа полнораундового блочного шифра ГОСТ 28147-89 с нулевой трудоемкостью и памятью. In: Доклад на конференции РусКрипто-2010.
[21] Гулак Г., Горбенко И., Михайленко М., Гитис Ю., 2003. Блочный симметричный криптоалгоритм SHACAL-2. Научно-технический сборник «Правовое, нормативное и метрологическое обеспечение», (7), с. 86-100.
[22] Гулак Г.М., 2008. Оцінка ризиків у ході проведення інженерного аналізу безпеки стеганографічних систем. Сборник научных трудов НАУ. Специальный выпуск , К.: НАУ, с. 259-264
[23] Гулак, Г. та Ковальчук, Л., 2001. Різні підходи до визначення випадкових послідовностей. Науково-технічний збірник «Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні», (3), с.127-133.
[24] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, 1(1):11-33, Jan-March 2004..
[25] В. Масловский, *Метод гарантированной защиты информации от утечки по каналам ПЭМИН*. Санкт-Петербург: Автореферат диссертации на соискание ученой степени кандидат технических наук и по ВАК РФ 05.13.19, 2003, с. 20.
[26] *ISO/IEC/IEEE 24765:2017 Systems and software engineering – Vocabulary*. [Online]. Available: https://www.iso.org/standard/71952.html. [Accessed: 06- Nov- 2020].

[27] П. Павленко, С. Філоненко та К. Бабіч, *Інформаційні системи і технології: навчальний посібник*. К.: НАУ, 2013, с. 324.

[28] В. Харченко, "Гарантоспособность и гарантоспособные системы: элементы методологии", *Радіоелектронні і комп'ютерні системи*, № 5, с. 7–19, 2006. [Accessed 24 December 2020].

[29] Федухин А.В., Сеспедес Гарсия Н.В., "Атрибуты и метрики гарантоспособных компьютерных систем", *Математичні машини і системи*, № 2, с. 195-201, 2013.

[30] Глухов В., "Оцінювання гарантоздатності криптографічних комп'ютерних систем", *Вісник Національного університету "Львівська політехніка"*, № 616, с. 66-72, 2008.

[31] Закон України Про електронні довірчі послуги / Відомості Верховної Ради (ВВР), 2017, № 45, ст. 400.

[32] І. Горбенко та Ю. Горбенко, *Прикладна криптологія: Теорія. Практика. Застосування: Монографія.*, 2nd ed. Харків: «Форт», 2012, с. 880.

[33] В. Ященко, Н. Варновский та Ю. Нестеренко, *Введение в криптографию*. М.: МЦНМО, 2012, с. 348.

[34] А. Бабаш та Г. Шанкин, *Криптография.* М.: СОЛОН-Р, 2002, с. 512

КІБЕРБЕЗПЕКА: освіта, наука, техніка
CYBERSECURITY: EDUCATION, SCIENCE, TECHNIQUE
№ 2 (10), 2020
ISSN 2663 - 4023

Hennadii M. Hulak
Ph.D Technical Sciences, Head of Laboratory Research Department №235
Institute of Mathematical Machines and Systems Problems, Kyiv, Ukraine
ORCID: 0000-0001-9131-9233
h.hulak@ukr.net

Volodymyr L. Buriachok
DSc in Technical Sciences, Professor, Head of the Department of Information and Cyber Security
Borys Grinchenko Kyiv University, Kyiv, Ukraine
ORCID ID: 0000-0002-4055-1494
v.buriachok@kubg.edu.ua

Pavlo M. Skladannyi
Senior Lecturer of the Department ofInformation and Cyber Security
Borys Grinchenko Kyiv University, Kyiv, Ukraine
ORCID: 0000-0002-7775-6039
p.skladannyi@kubg.edu.ua

Lydia V. Kuzmenko
methodist of the II category of the Center for Perspective Planning and Monitoring of Educational Activities of the National Academy of the Security Service of Ukraine, Kyiv
ORCID: 0000-0001-7392-0324
lido4ok@gmail.com

CRYPTOVIROLOGY: SECURITY THREATS TO GUARANTEED INFORMATION SYSTEMS AND MEASURES TO COMBAT ENCRYPTION VIRUSES

Анотація. This paper examines the security threats to guaranteed information systems, as well as measures to combat encryption viruses. A typical sequence of cyberattacks with information encryption using software tools to implement attacks is determined. The sequence of procedures of the WannaCry encryption part is described. The paper proposes a description of the computational complexity of encrypted data recovery problems, including methods for parallelizing the solution of cryptanalysis problems, methods used to solve problems of cryptosystem stability assessment, vulnerability search and decryption depending on basic mathematical methods. The application of distributed computing technology to solve problems of recovery of encrypted resources is described. The paper states that in modern conditions of constant development of cryptography methods with a high level of stability and their wide availability a necessary condition for improving the recovery of encrypted programs - data seekers is to create a special software security monitor and build specialized multiprocessor systems to implement cryptanalysis methods with wide access of authorized users. , from the point of view of minimizing the ratio "cost - the result" the most rational approach to creating specialized multiprocessor systems is to build a cluster system based on the most powerful general-purpose computers using hardware computing accelerators based on programmable logic integrated circuits to increase the effectiveness of attacks on cryptographic software tools for the implementation of attacks, it is advisable to develop technologies for active actions in cyberspace, in particular, those that provide the formation of hidden channels..

Ключові слова: cryptovirology, guarantee information systems, cyberattack, encryption, cryptography, cryptanalysis, cryptoalgorithm, programmable logic integrated circuits, software tools for attacks..

REFERENCES

[1] A. Young, M. Yung Cryptovirology: extortion-based security threats and countermeasures, Published in: Proceedings 1996 IEEE Symposium on Security and Privacy, 6-8 May 1996, Oakland, CA, USA, Publisher: IEEE DOI: 10.1109 /SECPRI.1996.502676 ISBN: 0-8186-7417-2, Pp. 125-140.

[2] A. Young, M. Yung. Malicious cryptography exposing cryptovirology. Wiley Publishing, Inc., p. 392, 2004.

[3] Clark., Z., 2017. The Worm That Spreads Wanacrypt0r. [online] Malwarebytes. Available at: <https://blog.malwarebytes.com/threat-analysis/2017/05/the-worm-that-spreads-wanacrypt0r/> [Accessed 5 October 2020].

[4] Suiche M., 2017. WannaCry Decrypting files with WanaKiwi + Demos. [online] Comae Technologies. Available at: https://blog.comae.io/wannacry-decrypting-files-with-wanakiwi-demo-86bafb81112d. [Accessed 5 October 2020].

[5] S.A. Shivale, "Cryptovirology: Virus Approach", International Journal of Network Security & Its Applications (IJNSA), № 3 (4), p.33-46, 2011.

[6] Kaur, J., Jaafar F. and Zavarsky P., 2018. An Empirical Analysis of Crypto-Ransomware Behavior. In: The Thirteenth International Conference on Systems (ICONS 2018). pp.1-7.

[7] Gorbenko ID, Gorbenko YI Applied cryptology: Theory. Practice. Application. Monograph. - Kharkiv, FORT, 2012, 880 p.

[8] Babash AV, Shankin GP, Cryptography, M .: SOLON-R, 2002, -512p.

[9] Bauer F. Decrypted Secrets: methods and maxims of cryptology. NY: Springer, p. 472, 1997.

[10] Schneier B. Applied cryptography. Protocols, algorithms, source texts in C language. - M .: Триумф, 2002. - 816 с.

[11] Alan G. Konheim, Computer Security and Cryptography, Published by John Wiley & Sons, Inc., Hoboken, New Jersey, p. 521, 2007.

[12] Grushko, A., 1999. On the existence of hidden channels. Discrete Mathematics, 1 (11), pp.24-28.

[13] Rowland C. H. Covert channels in the TCP / IP protocol suite. 1997, First Monday, 2 (5). https://doi.org/10.5210/fm.v2i5.528

[14] J. Kelsey Side Channel Cryptanalysis of Product Ciphers / J. Kelsey, B. Schneier, D. Wagner, C. Hall // 5th European Symposium on Research in Computer Security Louvain-la-Neuve, Belgium September 16–18, 1998 Proceedings, Berlin, Springer, 1998, pp.97-111.

[15] Shcherbakov A.Yu., Domashev AV, Applied cryptography. Use and synthesis of cryptographic interfaces. - M .: Publishing and trading house "Russian edition", 2003, 416 p .: ill.

[16] Gulak GM, Ensuring the security of CCI in cyberspace. Proceedings of the scientific and technical conference "Modern information and telecommunication technologies", volume IU Modern technologies of information security, -K .: - 2015, p.100-102.

[17] Gold B., Raider C. Digital signal processing. - M.,: «Sov. Radio », 1973, p. 368.

[18] Gorbenko I., Gulak G., Oleynikov R., Shumov A., Gorbenko Yu. 2003, Basic design principles, assessment of durability and prospects of using the AES encryption algorithm in Ukraine // Scientific and technical collection "Legal, regulatory and metrological support" ", (7), p.14

[19] Gorbenko ID, Gulak GM, Oliynikov RV, Ruzhentsev VI, Mikhailenko MS, 2005, Analysis of the properties of block symmetric encryption algorithms (according to the results of the international project NESSIE) // 8 International scientific-practical conference "Information security in information and telecommunication systems". Abstracts of reports. –K. 2005. p. 17-18.

[20] Rudskoy, V., On the zero practical significance of “Key determination attack of a full-round block cipher GOST 28147-89 with zero labor intensity and memory. In: Report at the RusCrypto-2010 conference.

[21] Gulak G., Gorbenko I., Mikhailenko M., Gitis Yu., 2003. Block symmetric cryptoalgorithm SHACAL-2. Scientific and technical collection "Legal, regulatory and metrological support", (7), p. 86-100.

[22] Gulak GM, 2008. Risk assessment during engineering safety analysis of steganographic systems. Collection of scientific works of NAU. Special issue, K .: NAU, p. 259-264.

[23] Gulak, G. and Kovalchuk, L., 2001. Different approaches to determining random sequences. Scientific and technical collection "Legal, regulatory and metrological support of the information protection system in Ukraine", (3), p.127-133.

[24] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. IEEE Transactions on Dependable and Secure Computing, 1 (1): 11-33, Jan-March 2004 ..

[25] V. Maslovsky, Method of guaranteed protection of information from leakage through PEMIN channels. St. Petersburg: Abstract of the thesis for the degree of candidate of technical sciences and HAC RF 05.13.19, 2003, p. twenty.
[26] ISO / IEC / IEEE 24765: 2017 Systems and software engineering - Vocabulary. [Online]. Available: https://www.iso.org/standard/71952.html. [Accessed: 06-Nov-2020].
[27] P. Pavlenko, S. Filonenko and K. Babich, Information Systems and Technologies: A Master Book. K .: NAU, 2013, p. 324.
[28] V. Kharchenko, "Dependability and Dependable Systems: Elements of Methodology", Radioelectronic and Computer Systems, No. 5, p. 7-19, 2006. [Accessed 24 December 2020].
[29] Fedukhin AV, Cespedes Garcia NV, "Attributes and metrics of reliable computer systems", Mathematical machines and systems, no. 195-201, 2013.
[30] V. Glukhov, "Assessment of the guaranteed publishing of cryptographic computer systems", Bulletin of the National University "Lvivska Politekhnika", no. 616, p. 66-72, 2008.
[31] Law of Ukraine On electronic servants / Vidomosty Verkhovnoy Radi (VVR), 2017, No. 45, art. 400.
[32] I. Gorbenko and Yu. Gorbenko, Applied Cryptology: Theory. Practice. Stuck: Monograph., 2nd ed. Kharkiv: "Fort", 2012, p. 880.
[33] V. Yashchenko, N. Varnovsky and Y. Nesterenko, Introduction to cryptography. M .: MTsNMO, 2012, p. 348.
[34] A. Babash and G. Shankin, Cryptography. M .: SOLON-R, 2002, p. 512