

## УДОСКОНАЛЕННЯ ЗАСТОСУВАННЯ АЛГОРИТМУ КРИПТОГРАФІЧНОГО ПЕРЕТВОРЕННЯ ГОСТ 28147-89 В РЕЖИМАХ ГАМУВАННЯ ТА ГАМУВАННЯ ЗІ ЗВОТНІМ ЗВ'ЯЗКОМ

В даній статті представлені пропозиції щодо удосконалення застосування алгоритму криптографічного перетворення, який описаний в ГОСТ 28147-89 “Системи обробки інформації. Захист криптографічний. Алгоритм криптографічного устаткування” в різних режимах гамування для передачі текстової, графічної та аналогової інформації. Удосконалений алгоритм може бути використаний для захисту інформації в автоматизованих системах управління військового призначення. Враховуючи це повстала проблема – якомога більше відокремити людину від керування ключами та захистити їх на всіх стадіях існування. У зв'язку з цим, нижче буде описано рішення цієї проблеми, за допомогою якого ГОСТ можна зробити теоретично недешифруємим.

**Ключові слова:** алгоритм криптографічного перетворення, синхропосилка, криптостійкість.

### Вступ

Методи криптографічного захисту інформації передбачають як програмне, так і апаратне шифрування. Програмна реалізація шифрування дешевше та практичніше. Тому, криптографічний стандарт ГОСТ 28147-89 (надалі ГОСТ) доцільно використовувати для захисту інформації, яка циркулює в комп'ютерних мережах або з'єднаних окремих ПЕОМ.

Та не зважаючи на це, система захисту в цілому не може бути надійнішою окремих її компонентів [1]. А компонентами такої системи будуть людина, програмний продукт, ПЕОМ, канал зв'язку (лінія з'єднання). Для того, щоб зламати систему, необхідно зламати один з її найслабкіших компонентів. Самий ненадійний ланцюг системи – людина. Бо користувач володіє ключами – а це є стержнева основа криптостійкості будь-якої криптографічної системи.

### Постановка задачі

Криптосхема, яка реалізує шифрування в режимі гамування представлена на рис. 1.

Криптосхема, приведена на рис. 1, містить [2]:

- ключовий запам'ятовуючий пристрій (КЗП) на 256 біт, що складається з восьми 32-розрядних накопичувачів ( $X_0, X_1, X_2, X_3, X_4, X_5, X_6, X_7$ );
- чотири 32-розрядних накопичувачі ( $N_1, N_2, N_3, N_4$ );
- два 32-розрядних накопичувачі ( $N_5, N_6$ ) із записаними в них постійними значеннями  $C_2, C_1$  (константи);
- два 32-розрядних суматора за модулем  $2^{32}$  ( $CM_1, CM_3$ );
- 32-розрядний суматор порозрядного підсумовування за модулем 2 ( $CM_2$ );
- 32-розрядний суматор за модулем  $(2^{32}-1)$  ( $CM_4$ );
- суматор за модулем 2 ( $CM_5$ ), обмеження на розрядність суматора  $CM_5$  не накладається;
- блок підстановки (К);
- регістр циклічного зрушення на одинадцять кроків убік старшого розряду (R).

Блок підстановки К складається з восьми вузлів заміни  $K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_8$  з пам'яттю на 64 бита кожний.

Ключовою системою в ГОСТі в режимі гамування є:

- ключ довжиною 256 біт, який знаходиться в КЗП (ключовий запам'ятовуючий пристрій);
- блок підстановок К (довгостроковий ключ, який можна представити у вигляді комутатора 32 x 32);
- синхропосилка 64 біт (сеансовий ключ).

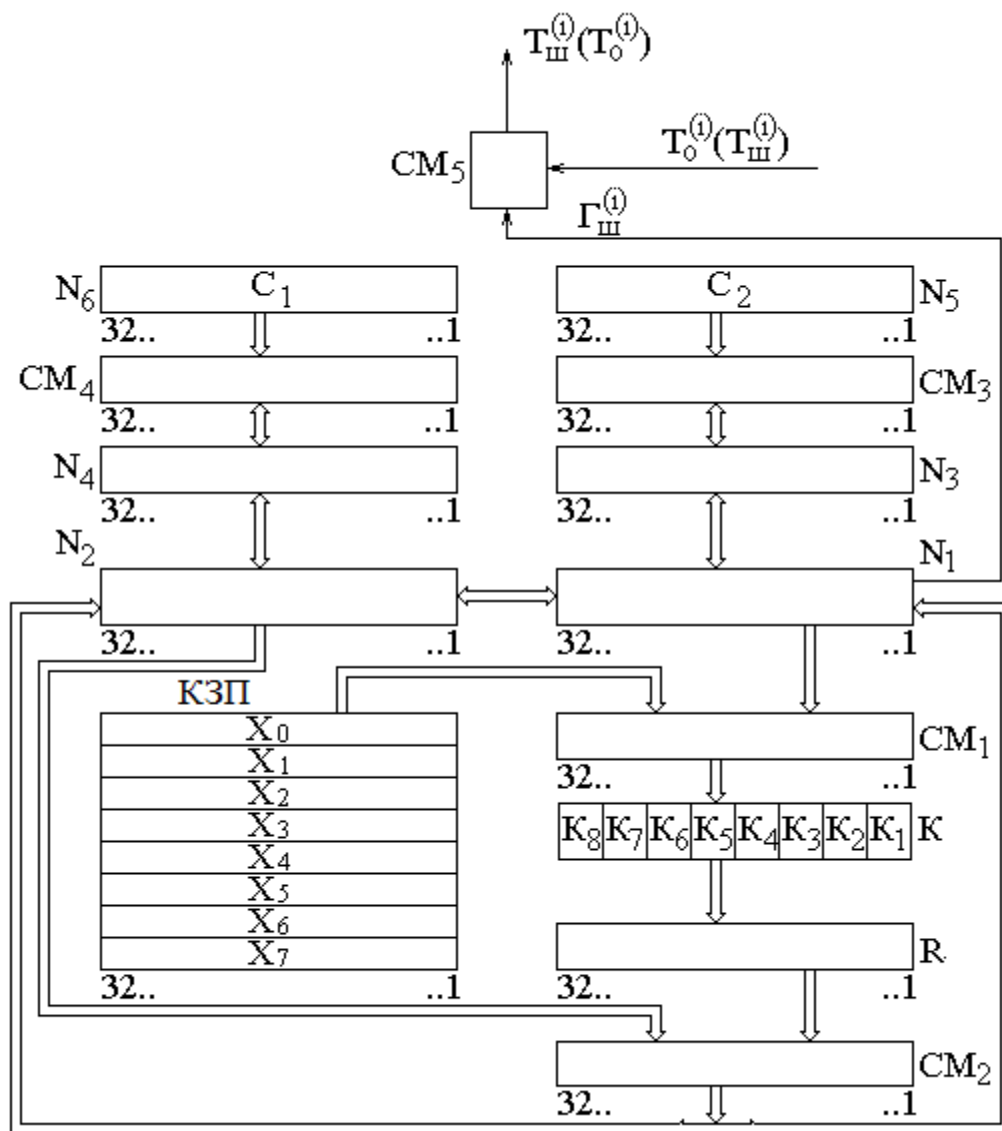


Рис.1. Криптосхема алгоритму ГОСТ 28147-89 в режимі гамування.

### Обмеження

1. Вихідний текст можливо відтворити із зашифрованого тексту лише за допомогою ключа дешифрування [3, 4];
2. Послідовне перебирання можливих ключів дешифрування, з метою відтворення вихідного тексту, потребує незоро великого часу обчислень або дуже великих затрат на реалізацію цих обчислень, та час на дешифрування завжди перевищує час “старіння” зашифрованої інформації;
3. Число можливих діючих ключів повинно бути непереборно великим (межа числа елементарних обчислень дорівнює  $10^{70}$ );
4. Інформація про алгоритм шифрування не повинна впливати на стійкість до злому системи шифрування;
5. Виключення читання “назад” (при захопленні криптографічної системи з ключами);
6. Статистика повідомлення повинна бути в значній мірі виключена зі статистики криптограм;
7. Незначна зміна ключа шифрування повинна приводити до суттєвих змін криптограми одного і того ж тексту;

8. Незначна зміна вихідного тексту повинна приводити до суттєвих змін криптограми в разі використання одного і того ж ключа;
9. Структурні елементи алгоритму шифрування повинні бути незмінними;
10. Додаткові біти, які вводять у повідомлення в процесі шифрування, повинні бути надійно замасковані в зашифрованому тексті;
11. Довжина зашифрованого повідомлення не повинна бути більшою, ніж саме повідомлення;
12. Не повинно бути простих залежностей між ключами, які послідовно використовують під час шифрування;
13. Стійкість шифру повинна зберігатися й у тому випадку, коли відомі деякі частини відкритого повідомлення, що передавалося, відповідно прийнятій криптограмі.

### Основна частина

#### Шифрування відкритих даних у режимі гамування:

Відкриті дані, розбиті на 64-розрядні блоки  $T_0^{(1)}, T_0^{(2)}, \dots, T_0^{(M-1)}, T_0^{(M)}$ , зашифровуються в режимі гамування шляхом порозрядного додавання по модулю 2 у суматорі  $CM_5$  з гамою шифру  $\Gamma_{ш}$ , що виробляється блоками по 64 бита, тобто

$$\Gamma_{ш} = (\Gamma_{ш}^{(1)}, \Gamma_{ш}^{(2)}, \dots, \Gamma_{ш}^{(M-1)}, \Gamma_{ш}^{(M)}),$$

де  $M$  – визначається обсягом шифруємих даних.

$\Gamma_{ш}^{(i)}$  –  $i$ -й 64-розрядний блок,  $i=1 \div M$ , число двійкових розрядів у блоці  $T_0^{(M)}$  може бути менше 64, при цьому невикористана для шифрування частина гами шифру з блоку  $\Gamma_{ш}^{(M)}$  відкидається.

У КЗП вводяться 256 біт ключа. У накопичувачі  $N_1, N_2$  вводиться 64-розрядна двійкова послідовність (синхропосилка)  $S=(S_1, S_2, \dots, S_{64})$ , яка є вихідним заповненням цих накопичувачів для наступного вироблення  $M$  блоків гами шифру. Синхропосилка вводиться в  $N_1$  і  $N_2$  так, що значення  $S_1$  вводиться в 1-й розряд  $N_1$ , значення  $S_2$  вводиться в 2-й розряд  $N_1$ , і т.д., значення  $S_{32}$  вводиться в 32-й розряд  $N_1$ ; значення  $S_{33}$  вводиться в 1-й розряд  $N_2$ , значення  $S_{34}$  вводиться в 2-й розряд  $N_2$  і т.д., значення  $S_{64}$  вводиться в 32-й розряд  $N_2$ .

Вихідне заповнення накопичувачів  $N_1$  і  $N_2$  (синхропосилка  $S$ ) зашифровується в режимі простої заміни. Результат шифрування переписується в 32-розрядні накопичувачі  $N_3$  і  $N_4$  так, що заповнення  $N_1$  пересилається в  $N_3$ , а заповнення  $N_2$  пересилається в  $N_4$ .

Заповнення накопичувача  $N_4$  сумується по модулю  $(2^{32}-1)$  у суматорі  $CM_4$  з 32-розрядною константою  $C_1$  з накопичувача  $N_6$ , результат записується в  $N_4$ .

Заповнення накопичувача  $N_3$  сумується по модулю  $2^{32}$  у суматорі  $CM_3$  з 32-розрядною константою  $C_2$  з накопичувачем  $N_5$ , результат записується в  $N_3$ .

Заповнення  $N_3$  пересилається в  $N_1$ , а заповнення  $N_4$  пересилається в  $N_2$ , при цьому заповнення  $N_3, N_4$  зберігається.

Заповнення  $N_1$  і  $N_2$  зашифровується в режимі простої заміни. Отримане в результаті шифрування заповнення  $N_1, N_2$  утворює перший 64-розрядний блок гами шифру  $\Gamma_{ш}^{(1)}$ , що сумується порозрядно по модулю 2 у суматорі  $CM_5$  з першим 64-розрядним блоком відкритих даних  $T_0^{(1)}=(t_1^{(1)}, t_2^{(1)}, \dots, t_{63}^{(1)}, t_{64}^{(1)})$ ... У результаті підсумовування виходить 64-розрядний блок зашифрованих даних  $T_{ш}^{(1)}=(\tau_1^{(1)}, \tau_2^{(1)}, \dots, \tau_{63}^{(1)}, \tau_{64}^{(1)})$ ...

Значення  $\tau_1^{(1)}$  блоку  $T_{ш}^{(1)}$  є результатом підсумовування по модулю 2 у  $CM_5$  значення  $t_1^{(1)}$  із блоку  $T_0^{(1)}$  зі значенням 1-го розряду  $N_1$ , значення  $\tau_2^{(1)}$  блоку  $T_{ш}^{(1)}$  є результатом підсумовування по модулю 2 у  $CM_5$  значення  $t_2^{(1)}$  із блоку  $T_0^{(1)}$  зі значенням 2-го розряду  $N_1$  і т.д., значення  $\tau_{64}^{(1)}$  блоку  $T_{ш}^{(1)}$  є результатом підсумовування по модулю 2 у  $CM_5$  значення  $t_{64}^{(1)}$  із блоку  $T_0^{(1)}$  зі значенням 32-го розряду  $N_2$ .

Для одержання наступного 64-розрядного блоку гами шифру  $\Gamma_{ш}^{(2)}$  заповнення  $N_4$  сумується по модулю  $(2^{32}-1)$  у суматорі  $CM_4$  з константою  $C_1$  з  $N_6$ , заповнення  $N_3$  сумується по модулю  $2^{32}$  у суматорі  $CM_3$  з константою  $C_2$  з  $N_5$ . Нове заповнення  $N_3$  пересилається в  $N_1$ , а нове заповнення  $N_4$  пересилається в  $N_2$ , при цьому заповнення  $N_3$  і  $N_4$  зберігається.

Заповнення  $N_1$  і  $N_2$  зашифровується в режимі простої. Отримане в результаті шифрування заповнення  $N_1$ ,  $N_2$  утворить другий 64-розрядний блок гами шифру  $\Gamma_{ш}^{(2)}$ , що сумується порозрядно по модулю 2 у суматорі  $СМ_5$  із другим блоком відкритих даних  $T_0^{(2)}$ . Аналогічно виробляються блоки гами шифру  $\Gamma_{ш}^{(3)}$ ,  $\Gamma_{ш}^{(4)}$  ...,  $\Gamma_{ш}^{(M)}$  і зашифровуються блоки відкритих даних  $T_0^{(3)}$ ,  $T_0^{(4)}$  ...,  $T_0^{(M)}$ . Якщо довжина останнього  $M$ -го блоку відкритих даних  $T_0^{(M)}$  менше 64 біт, то з останнього  $M$ -го блоку гами шифру  $\Gamma_{ш}^{(M)}$  для шифрування використовується тільки відповідне число розрядів гами шифру, інші розряди відкидаються.

**Розшифрування зашифрованих даних у режимі гамування:**

При розшифруванні криптосхема має той же вид, що і при шифруванні (див. рис. 1). У КЗП вводяться 256 біт ключа, за допомогою якого здійснювалося шифрування даних  $T_0^{(1)}$ ,  $T_0^{(2)}$  ...,  $T_0^{(M)}$  при цьому  $T_0^{(M)}$  може містити менше 64 розрядів.

Звичайно, що для розшифрування необхідна одна і та ж синхропосилка, тому у канал зв'язку передається синхропосилка  $S$  і блоки зашифрованих даних  $T_{ш}^{(1)}$ ,  $T_{ш}^{(2)}$  ...,  $T_{ш}^{(M)}$ . Із вимоги унікальності гами, невиконання якої приводить до катастрофічного зниження стійкості шифру, впливає, що для шифрування двох різних масивів даних на одному ключі необхідно забезпечити використання різних синхропосилок. Це призводить до необхідності зберігати або передавати синхропосилку каналом зв'язку (що суттєво зменшує стійкість шифрування).

На перший погляд алгоритм криптографічного стандарту ГОСТ 28147-89 є бездоганним, але як застосовувати цей алгоритм в ГОСТі не визначено. Хоча, методика застосування, того, чи іншого криптографічного алгоритму, значно впливає на зменшення або підвищення його стійкості.

Тому пропонується:

1. Синхропосилку виробляти автоматично за методом “залікового відрізка” використовуючи для цього лінійні рекурентні регістри.
2. В криптосхему ГОСТа додати ще один накопичувач інформації ємністю 256 біт, який би за відповідним алгоритмом накопичував разову похідну ключа, а після закінчення кожного 32 циклу алгоритму шифрування, змінював ключове заповнення в КЗП стираючи попереднє.

**Алгоритм реалізації**

**Псевдовипадкові послідовності, які змінюються від одного перезапуску до другого:**

В ролі синхропосилки пропонується використовувати рекурентні послідовності. Схема, яка реалізує цю ідею представлена на рис. 2. [2, 5]

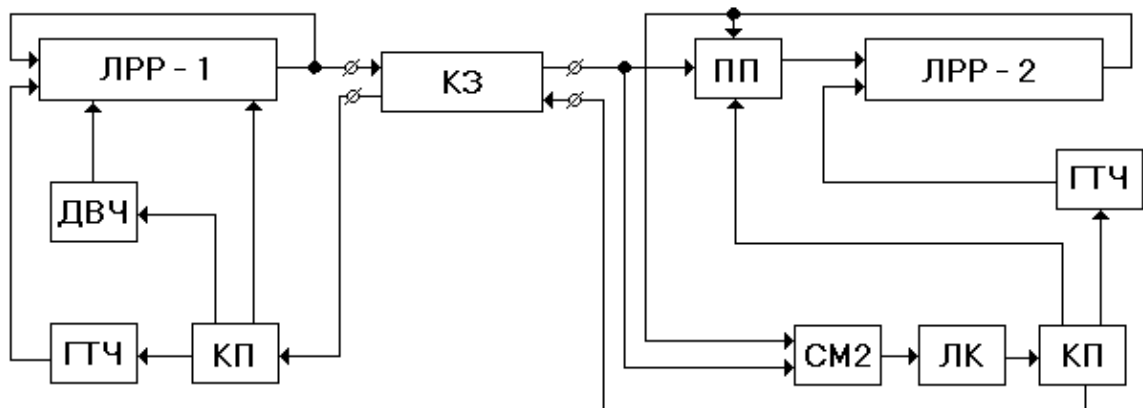


Рис.2. Схема виробітки синхропосилки по методу “залікового відрізка”.

На передаючій стороні вона складається з лінійного рекурентного регістру ЛРР-1 із зворотнім зв'язком, генератора шуму або датчику випадкових чисел (ДВЧ), генератору тактової частоти (ГТЧ), керуючого пристрою (КП). На приймальній стороні схема включає в

себе ЛРР-2, аналогічний ЛРР-1, генератор тактової частоти (ГТЧ), аналізатор у вигляді суматору по модулю 2 (СМ2), лічильник (ЛК), керуючий пристрій (КП), перемикаючий пристрій (ПП).

Для запуску системи на ЛРР-1 з ДВЧ подається будь-яка випадкова послідовність одиниць та нулів, що заповнює чарунки пам'яті регістру, пройшовши всі чарунки, ця ж послідовність подається у канал зв'язку (КЗ). Після заповнення ЛРР-1 початковим заповненням, ДВЧ відключається, а інформація в ЛРР-1 циркулює по замкненому колу, паралельно передаючись у КЗ.

Приймальна сторона знаходиться в режимі чекання. Коли на вхід системи поступає послідовність двійкової інформації, вона поступово записується у ЛРР-2 та паралельно подається на вхід аналізуючого суматору СМ2. Доки на ЛРР-2 не буде задіяний зворотній зв'язок, СМ2 видає непередбачувану двійкову послідовність, при кожній одиниці лічильник ЛК повертається у нульове положення.

Після заповнення всіх  $n$  чарунок ЛРР-2 елементами синхросилки, місцева рекурентна послідовність стає ідентичною з отриманою рекурентною послідовністю з каналу зв'язку (за умови відсутності помилок у КЗ). При цьому схема поелементного аналізу видає серію нулів.

Якщо  $k$  разів підряд буде мати місце співпадання елементів синхросилки з елементами місцевої рекурентної послідовності, то на виході лічильника з'явиться сигнал, який поступить до керуючого пристрою, сигналізуючи про те, що системи працюють синфазно і синхронно. Перед цим лічильник відраховує певну кількість нулів, яку ми вибираємо, як "заліковий відрізок".

КП дає команду на ПП, який відключає ЛРР-2 від КЗ та переводить його в автономний режим роботи, перемикаючи ланцюг зворотного зв'язку. КП дає команду на передаючу сторону про зупинку ЛРР-1. Після зупинки ЛРР-1,

ЛРР-2 робить ще  $n$  тактів (де  $n$  – максимальна ступінь поліному за яким було побудовано ЛРР), в результаті ми маємо в ЛРР-1 і ЛРР-2 однакові заповнення.

Знаючи значення ЛРР-1, що передається по КЗ, можливо частково відновити початкове заповнення ЛРР-1 та ЛРР-2. Таким чином можливо дізнатися про частину разового ключа. Щоб запобігти розкриттю синхросилки застосовується перетворення наповнення ЛРР-1 і ЛРР-2 на певну кількість циклів, однакову на обох сторонах. Ця кількість задається контрольною сумою ключа який записано в КЗП.

Після повних перетворень заповнення ЛРР-1 та ЛРР-2 є однаковими, виробленими автоматично, невідомі ні криптоаналітику ні користувачу, і можуть бути використані у якості синхросилки.

Для цього ЛРР-1 та ЛРР-2 побудовані на неприводимих, примітивних поліномах 64 ступеню.

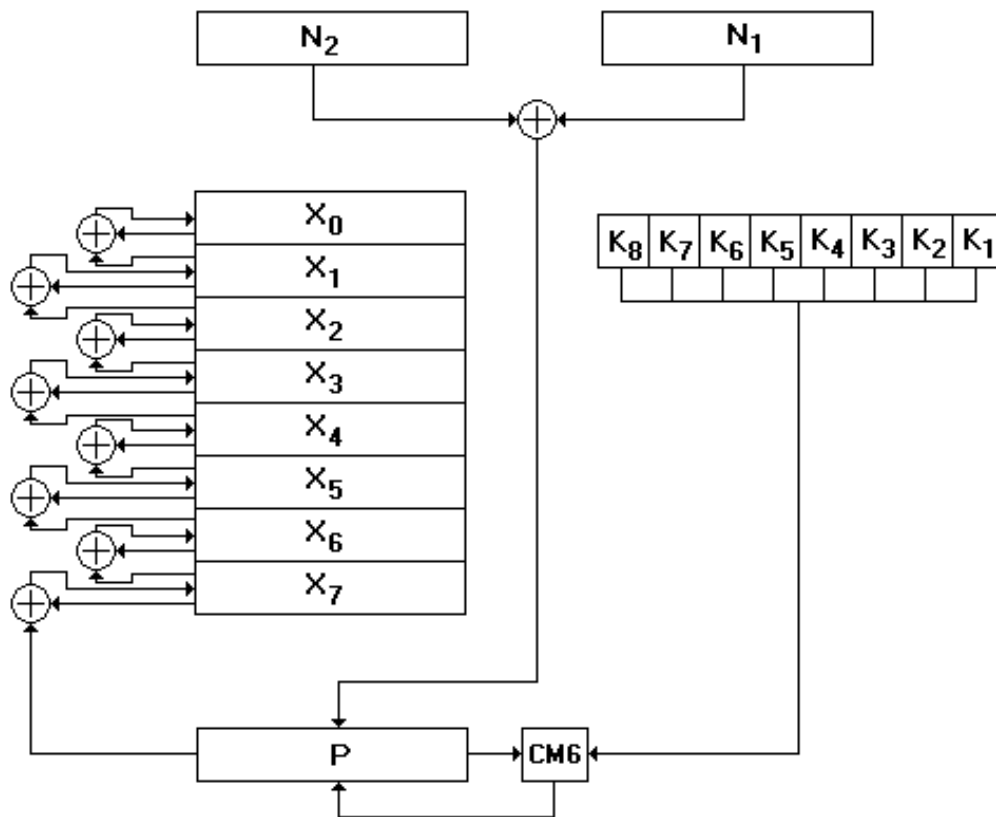
#### **Формування нових разових похідних ключових даних із проміжних гам:**

У зв'язку з тим, що ключові дані (КД) в криптосхемі алгоритму ГОСТ знаходяться в КЗП – це створює передумови для "читання назад". Однак від цього можливо позбавитися шляхом періодичного формування нових ключів із проміжних гам, і використовувати ці ключі в якості нових даних для КЗП. Період поновлення КД повинен бути вибраним таким чином щоби "читання назад" на цьому інтервалі часу не представляло серйозної небезпеки. [2, 6]

Разові похідні ключових даних виробляються автоматично без втручання користувача. Через певну кількість оброблених блоків інформації, разова похідна повністю замінює ключ, який зберігається в КЗП.

Для вироблення разової похідної ключа (рис. 3) використовується допоміжний 32-розрядний накопичувач  $P$ . На початку роботи накопичувач  $P$  містить двійкову послідовність як суму за модулем 2 двох частин синхросилки  $S_1$  і  $S_2$ . В кожному циклі роботи криптосхеми, значення накопичувача  $P$  підсумовується за модулем 2 в СМ6 з 32

бітами двійкової інформації отриманої після блоку заміни К, після чого результат суми знову записується в накопичувач Р.



- де символ  $\oplus$  виконує функцію суматора за модулем 2.

Рис.3. Блок-схема формування ключових даних з проміжних гам.

Після 32-го циклу роботи криптосхеми здійснюється заміна ключа в КЗП за наступним алгоритмом:

Значення накопичувачів  $X_0$  і  $X_1$  з КЗП підсумовується за модулем 2. Результат записується в накопичувач  $X_0$ . Потім аналогічно підсумовується значення накопичувачів  $X_1$  і  $X_2$ ,  $X_2$  і  $X_3$ ,  $X_3$  і  $X_4$ ,  $X_4$  і  $X_5$ ,  $X_5$  і  $X_6$ ,  $X_6$  і  $X_7$ . Результати підсумовувань будуть записані у накопичувачі  $X_1$ ,  $X_2$ ,  $X_3$ ,  $X_4$ ,  $X_5$ ,  $X_6$  відповідно.

Значення накопичувача підсумовується за модулем 2 зі значенням накопичувача  $X_7$  із КЗП. Результат суми записується в накопичувач  $X_7$ .

В математичному вигляді всі ці операції можна записати наступним чином:

$$\begin{aligned} X_0 &= X_0 \oplus X_1; X_1 = X_1 \oplus X_2 \\ X_2 &= X_2 \oplus X_3; X_3 = X_3 \oplus X_4 \\ X_4 &= X_4 \oplus X_5; X_5 = X_5 \oplus X_6 \\ X_6 &= X_6 \oplus X_7; X_7 = X_7 \oplus P \end{aligned}$$

Враховуючи все вищезазначене, після 32-го циклу роботи криптосхеми пройде заміна даних у КЗП. Це свідчить про те, що новий 32 цикловий період виробки наступного 64-бітного блоку гами шифру почнеться на нових, автоматично вироблених КД.

При чому початкове заповнення КЗП зберігається для організації наступних сеансів зв'язку.

**Висновки та рекомендації:**

Таким чином, в результаті удосконалення застосування ГОСТу отримуємо наступні переваги:

- одна з трьох компонент ключа виробляється автоматично, навіть оператор не знає синхропосилки;
- через 32 цикли, автоматично змінюється основний ключ в КЗП і буде змінюватись кожні 32 цикли. Тобто, кожний 64 бітний блок відкритих даних шифруватиметься на своєму, особистому, автоматично виробленому ключі, без втручання людини;
- при перехваті криптограми та захваті ПЕОМ на якому ця криптограма оброблялася, однозначного дешифрування не відбудеться.

**Література**

1. Конхейм А. Г. Основи криптографії / А. Г. Конхейм. – М.: Радіо і зв'язок, 1987.
2. ГОСТ 28147-89 Системи обробки інформації. Захист криптографічний. Алгоритм криптографічного устаткування.
3. Шеннон К. Роботи по теорії інформації та кібернетики / К. Шеннон. – М.: ИЛ, 1969.
4. Хоффман Л. Сучасні методи захисту інформації / Л. Хоффман. – М.: Радянське Радіо, 1980.
5. Жельников В. Криптографія від папірусу до комп'ютера / В. Жельников. – М.: АБФ, 1996. – 336 с.
6. Романец В. Ю. Защита информации в компьютерных системах и сетях / В. Ю. Романец. – М.: Радио и связь, 2001. – 376 с.

Надійшла 06.02.2017 р.

Рецензент: к.т.н., доц. Гулак Г.М.