S E N S ' 2 0 0 6
Second Scientific Conference with International Participation
SPACE, ECOLOGY, NANOTECHNOLOGY, SAFETY
14 – 16 June 2006, Varna, Bulgaria

ГЕНЕРИРАНЕ НА КЛЮЧ С ИЗПОЛЗВАНЕ НА ЛИНЕЕН ПРЕМЕСТВАЩ РЕГИСТЪР С ОБРАТНА ВРЪЗКА В СИНХРОННИ И САМОСИНХРОНИЗИРАЩИ СЕ КРИПТОСИСТЕМИ С ПОТОЧНО ШИФРИРАНЕ

Адриана Бороджиева

Русе 7017, ул. „Студентска" № 8, Русенски университет „Ангел Кънчев",
Катедра „Комуникационна техника и технологии", тел.: (00359 82) 888 734,
e-mail: aborodjieva@ecs.ru.acad.bg

Ключови думи: криптосистеми, поточно шифриране, линейни преместващи регистри с обратна връзка.

Резюме. *Статията представя накратко модела на процесите на шифриране и дешифриране в една комуникационна система; посочва задачите, които трябва да се решават при проектирането на криптосистемите, както и причините за широкото им навлизане в комуникациите през последните години. Разгледани са по-често срещаните класически заплахи за разбиване на защитата на криптосистемите. В материала се описва подробно поточното шифриране и някои от използваните технологии за поточно шифриране. Представен е един пример за генериране на ключ с използване на линеен преместващ регистър с обратна връзка, като е обърнато специално внимание и на слабите места на този метод за генериране на ключ. Накрая се разглежда и структурата на синхронните и самосинхронизиращи се криптосистеми с поточно шифриране, като се посочват техните основни предимства и недостатъци.*

ВЪВЕДЕНИЕ

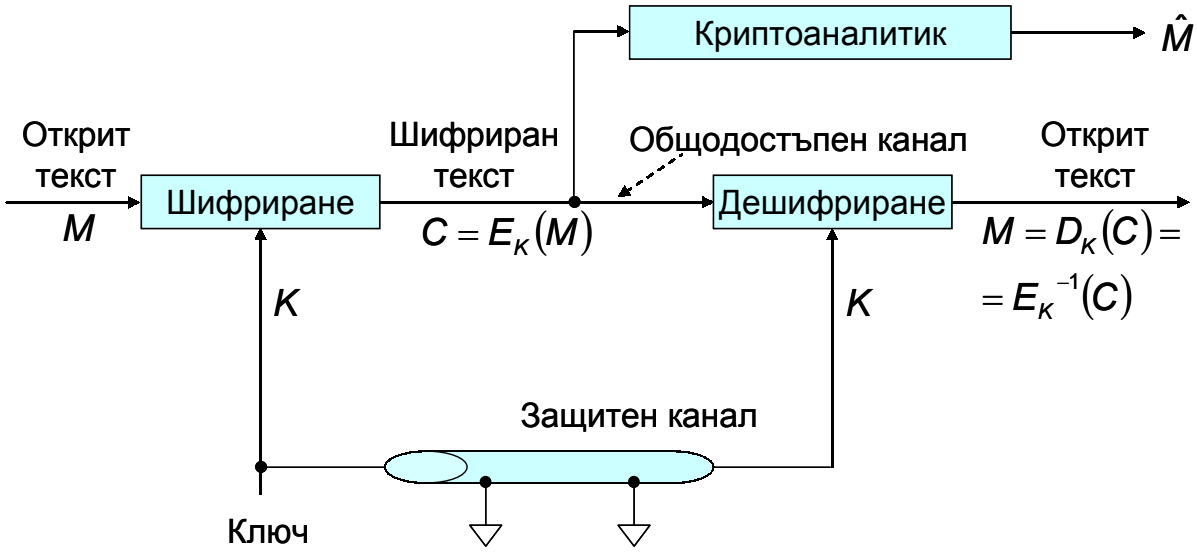
Желанието за конфиденциални комуникации крие своите дълбоки корени в далечното минало и се свързва с древните гърци и спартанци, както и с името на император Юлий Цезар. Още известният гръцки историк Херодот (от V в.пр.н.е.) е говорил за тайнописи, разбираеми само от получателя, за когото са предназначени, а спартанците са използвали специален механизъм за неразбираемо записване на важните заповеди и сведения. Император Юлий Цезар е изпращал секретни съобщения, заменяйки буквата *A* с буква *D*, буквата *B* с буква *E* и т.н., и само предварително запознатите с простата субституция "*shift by 3*" (*изместване на азбуката с 3*) са можели да разбират съдържанието на посланията му. Логично е да се предположи, че Юлий Цезар е използвал и по-общата субституция "*shift by n*" (*изместване на азбуката с n*) [1]. Изучаването на начина на предаване на съобщенията, които не допускат чуждо вмешателство, се нарича *криптография*. Терминът *шифриране* (*криптиране*) обозначава преобразуването на съобщението, изпълнявано от предавателя, а терминът *дешифриране* – обратното преобразуване, извършвано от приемника. Основните причини за използване на криптосистеми в комуникациите се явяват: (1) *обезпечаване на конфиденциалност*, което означава предотвратяване на извличането на информация от канала от странични лица (*подслушване*); (2) *автентификация*, т.е. предотвратяване на вмъкването на информация в канала от външни лица (*измамен достъп*). Например, много често в електронната поща трябва да се осигури електронният еквивалент на *подписа* с цел

отстраняване на всякакви недоразумения между изпращащия и получателя относно това какво съобщение е било изпратено и било ли е то изпратено въобще [2].

МОДЕЛ НА ПРОЦЕСИТЕ НА ШИФРИРАНЕ И ДЕШИФРИРАНЕ, ЗАДАЧИ НА КРИПТОСИСТЕМИТЕ И КЛАСИЧЕСКИ ЗАПЛАХИ

На фиг.1 е показан модел на криптографски канал [2]. Съобщението (откритият текст) M се шифрира с помощта на обратимото преобразуване E_K, при което се получава шифрирания текст $C = E_K(M)$. От своя страна, шифрираното съобщение C се пропуска през общодостъпен (незащитен) канал и след получаването му в приемника, неговото изходно значение може да се възстанови с помощта на операция *дешифриране*, описана с обратното преобразуване $D_k = E_k^{-1}$, което има следния вид:

(1) $D_k(C) = E_k^{-1}[E_k(M)] = M$.



Фиг.1. Модел на криптографски канал

Параметърът K обозначава множеството от символи или характеристики, наричани *ключове*, които определят конкретното шифриращо преобразуване E_K от семейството криптографски преобразувания. Първоначално защитеността на криптосистемите е зависила от секретността на целия процес на шифриране, но впоследствие са били разработени системи, за които общата природа на преобразуването при шифриране или алгоритъмът може да са общоизвестни, а секретността на системата да зависи от специален ключ. Ключът се използва, както за шифриране на нешифрирано съобщение, така и за дешифриране на шифрираното съобщение. Следователно, в повечето криптосистеми, всеки, който има достъп до ключа, може както да шифрира, така и да дешифрира съобщения. Ключът се предава чрез секретен канал на авторизирани потребители и по правило остава непроменен за много предавания. Целта на *криптоаналитика* (*противника*) се явява оценка на открития текст $\hat{M}$ чрез анализ на шифрирания текст, получен от общодостъпния канал, но без използване на ключа.

Схемите за шифриране могат да се разделят на две основни категории [1,2]: *блокови* и *поточни* (*за шифриране на потока от данни*). При блоковото шифриране откритият текст се разделя на блокове с фиксиран размер, след което всеки блок се шифрира независимо. Следователно, с помощта на зададения ключ, еднаквите блокове на открития текст ще се преобразуват в еднакви блокове на шифрирания текст. При поточното шифриране не съществуват блокове с фиксиран размер и

всеки бит от открития текст m_i се шифрира с помощта на i-тия елемент k_i от последователността на символите на генерирания ключов поток. Процесът на шифриране се явява *периодичен*, ако ключовият поток започне да се повтаря след p символа (при фиксирано p); в противен случай – *непериодичен*.

Основните изисквания към криптосистемите могат да се формулират по следния начин: (1) осигуряване на прости и нескъпи средства за шифриране и дешифриране за авторизирани потребители, притежаващи съответния ключ; (2) задачата на криптоаналитика по изработването на оценка на нешифрирания текст без помощта на ключа да бъде максимално сложна и скъпа [2].

Криптосистемите се делят на *безусловно защитени* или схеми, *защитени по изчисления*. Казва се, че една система е *безусловно защитена*, ако наличната за криптоаналитика информация не е достатъчна за определяне на преобразуванията при шифриране и дешифриране, независимо от изчислителната мощност, с която разполага той. Една такава система, наричана *система с еднократно запълване*, включва шифриране на съобщението с помощта на случаен ключ, който се прилага само един път. Тъй като ключът никога не се използва повторно, криптоаналитикът не притежава информация, която може да използва за разшифриране на следващото предавано съобщение. Въпреки че такава система се явява безусловно защитена, в съвременните комуникационни системи тя се прилага рядко, тъй като за всяко ново съобщение е необходимо да се разпространи новия ключ, а обикновено това е трудно. Въобще, разпределението на ключовете за авторизираните потребители се явява основен проблем при използването на произволна криптосистема, даже ако ключът се прилага в течение на продължителен период от време. Въпреки че може да се докаже безусловната защитеност на някои криптосистеми, в настоящия момент не съществува обща схема за доказателство на защитеността на произволна криптосистема. Всъщност, в спецификациите на повечето криптосистеми се указва формално, че те са *защитени по изчисления* за x години, което означава, че при благоприятни за криптоаналитика обстоятелства, т.е. при използване на най-съвременните компютри, защитата на системата може да бъде разбита за x години, но не и по-рано.

Най-незначителната криптоаналитическа заплаха е *атаката само с шифрирания текст* (*ciphertext-only attack*). В този случай криптоаналитикът може да притежава *някаква* информация за общата система и езика, използван в съобщението, но единствените важни данни, с които разполага той, се явява шифрираното съобщение, прихванато от общодостъпния канал.

По-сериозна заплаха за системите се явява *атаката с известен открит текст* (*known plaintext attack*), която включва познаване на открития текст и неговия шифриран еквивалент. Твърдата структура на повечето бизнес-форми и програмни езици често дава на опонента априорни знания за елементите на открития текст. "Въоръжен" с тези знания и шифрираното съобщение, криптоаналитикът може да проведе криптоанализ с помощта на известния открит текст.

Ако криптоаналитикът трябва да избере открит текст за дадено шифрирано съобщение, заплахата се нарича *атака с избран открит текст* (*chosen plaintext attack*). Такава атака е била използвана по време на Втората световна война от САЩ с цел получаване на повече информация за японската криптосистема [2].

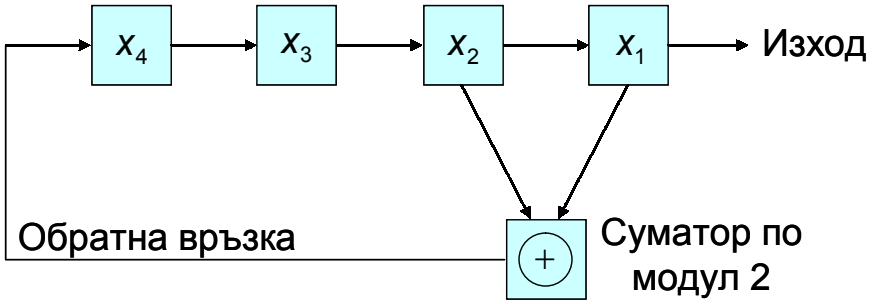
ПОТОЧНО ШИФРИРАНЕ

Еднократното запълване бе определено по-горе като система за шифриране със случаен еднократен ключ, осигуряващ безусловна защитеност. Реализирането на еднократно поточно запълване е възможно чрез използването на действително случаен поток от ключове, които никога не се повтарят. По такъв начин съвършената секретност може да се достигне за безкраен брой съобщения така, че всяко

съобщение се шифрира с помощта на различни части от случайния ключов поток. Схемите за поточно шифриране са опит за имитиране на едномоментно запълване, като основен етап в реализацията им е генерирането на случайни ключови потоци с помощта на съответни алгоритми. Най-популярната технология за поточно шифриране използва *псевдослучайни последователности*, чието наименование отразява факта, че те наистина изглеждат случайни за случайния наблюдател, но в същото време се явяват детерминистични. При тази технология алгоритмите за шифриране и дешифриране се реализират чрез използване на *преместващи регистри с обратна връзка*. Може да се покаже, че поточният псевдослучаен ключ осигурява същата защитеност, както и методът на едномоментно запълване, тъй като периодът на повторение на последователността, породена от линеен преместващ регистър, е $2^n - 1$ бита, където n е броят на разрядите в регистъра. Ако псевдослучайната последователност се реализира с помощта на 50-разряден регистър, при дискретизация 1MHz, тя ще се повтаря на всеки $2^{50} - 1$ микросекунди, или на всеки 35 години. В епохата на големите интегрални схеми не е трудно да се реализира схема със 100 разряда. В този случай последователността ще се повтаря на всеки 4×10^{16} години. Следователно, генерираната последователност може действително да се нарече случайна, тъй като не се повтаря в течение на такъв продължителен период от време, и тя ще гарантира съвършена секретност. Но все пак съществува едно важно отличие на псевдослучайната последователност от действително случайната, която се използва в метода на едномоментното запълване. Псевдослучайната последователност се генерира от алгоритъм и ако е известен алгоритъма, то е известна и самата последователност. Поради тази особеност, схемата за шифриране, която използва линеен преместващ регистър с обратна връзка, е твърде уязвима към *атака с известен открит текст*.

ПРИМЕР ЗА ГЕНЕРИРАНЕ НА КЛЮЧ С ИЗПОЛЗВАНЕ НА ЛИНЕЕН ПРЕМЕСТВАЩ РЕГИСТЪР С ОБРАТНА ВРЪЗКА

В технологията за поточно шифриране за генерирането на псевдослучайна ключова последователност обикновено се използват преместващи регистри. Те могат да бъдат превърнати в генератор на псевдослучайни последователности чрез въвеждане на контур за обратна връзка, който изчислява новата стойност за първия разряд, базирайки се на предишните стойности на n-те елемента. Ако в контура за обратна връзка се извършва линейна операция, то регистърът се нарича *линеен*. На фиг.2 е показан такъв генератор на псевдослучайна последователност при $n = 4$.



Фиг.2. Пример за линеен преместващ регистър с обратна връзка

В дадения пример разрядите на регистъра е удобно да се номерират така, както е показано на фиг.2, а изходите на елементите 1 и 2 се сумират по модул 2 (линейна операция) и се предават обратно към елемент 4. Ако началното състояние на разрядите (x_4, x_3, x_2, x_1) е 1000, то следващите състояния ще бъдат: 0100, 0010, 1001, 1100 и т.н. Изходната последователност се образува от битовете, снети от крайния десен елемент на регистъра, т.е. в случая тя ще бъде 111101011001000,

където крайният десен бит в редицата се явява най-ранно предаденият, а крайният ляв бит – най-късно предаденият бит. При даден произволен n-разряден линеен преместващ регистър с обратна връзка, изходната последователност се счита за периодична.

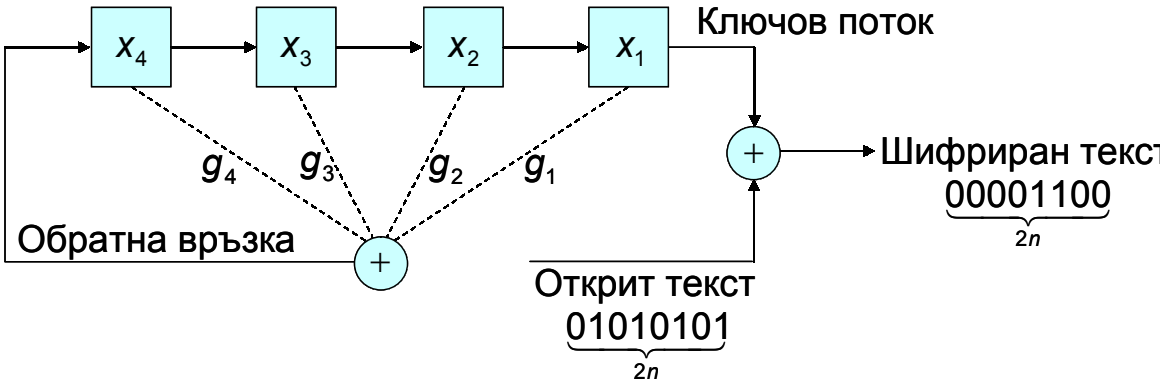
СЛАБИ МЕСТА НА ЛИНЕЙНИТЕ ПРЕМЕСТВАЩИ РЕГИСТРИ С ОБРАТНА ВРЪЗКА

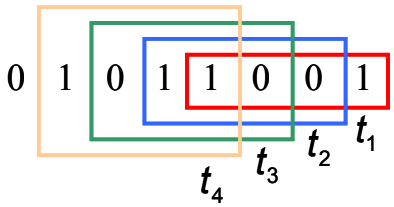
Схемата за шифриране, в която, за пораждане на ключовия поток, се прилагат *линейни преместващи регистри с обратна връзка* (*linear feedback shift register* – LFSR), се явява много уязвима по отношение на атаки [2]. За да се определят съединенията на обратната връзка, началното състояние на регистъра и цялата последователност на кода, криптоаналитикът се нуждае само от $2n$ бита на открития текст и от съответния им шифриран текст. Като правило, $2n$ е много по-малко от периода на повторение на последователността $2^n - 1$. Тази уязвимост ще се илюстрира с помощта на примерния регистър, показан на фиг.2. Нека криптоаналитикът не познава вътрешните връзки на регистъра, но му се отдава възможност да получи $2n = 8$ бита шифриран текст и неговия открит еквивалент.

Открит текст: 01010101 **Шифриран текст:** 00001100

Тук крайният десен бит е получен първи, а крайният ляв бит – последен.

За да получи фрагментът от ключовия поток 01011001, криптоаналитикът събира по модул 2 двете последователности (на открития и на шифрирания текст). Ключовият поток показва съдържанието на регистъра в различни моменти от време. Крайните десни четири ключови бита показват съдържанието на преместващия регистър в момент t_1. Ако последователно се "премества" тази четворка на един разряд вляво, ще се получава съдържанието на регистъра в моменти t_2, t_3 и t_4.





Фиг.3. Пример за уязвимостта на линеен преместващ регистър с обратна връзка

Използвайки линейната структура на преместващия регистър, може да се запише следната зависимост:

(2) $g_4 x_4 \oplus g_3 x_3 \oplus g_2 x_2 \oplus g_1 x_1 = x_5$.

Тук x_5 е цифра, която чрез контура на обратната връзка, се подава обратно на входа на регистъра, а g_i определя i-тото съединение с обратната връзка. По такъв начин, като се използва съдържанието на регистъра в четири момента от време (фиг.3), могат да се запишат следните четири уравнения с четири неизвестни:

$$
(3)\qquad
\begin{aligned}
&g_4.1 \oplus g_3.0 \oplus g_2.0 \oplus g_1.1 = 1\\
&g_4.1 \oplus g_3.1 \oplus g_2.0 \oplus g_1.0 = 0\\
&g_4.0 \oplus g_3.1 \oplus g_2.1 \oplus g_1.0 = 1\\
&g_4.1 \oplus g_3.0 \oplus g_2.1 \oplus g_1.1 = 0
\end{aligned}.
$$

Решавайки уравненията (3), криптоаналитикът узнава връзките на регистъра, а също и неговото начално състояние в момент t_1. Следователно, той може да узнае последователността в кой да е момент от време. В случая, за решение на уравнения (3) се получава $g_1 = 1, g_2 = 1, g_3 = 0, g_4 = 0$, като $g_i = 1$ определя наличието на съединение на i-тия елемент на регистъра с обратната връзка, а $g_i = 0$ – отсъствието на такова съединение. Обобщавайки този пример за всеки преместващ регистър с n разряда, уравнение (2) може да се запише по следния начин:

$$
(4)\qquad x_{n+1} = \sum_{i=1}^{n} g_i x_i \ \text{ по модул 2}
$$

или в матрична форма:

(5) $\mathbf{x} = \mathbf{X}\mathbf{g}$, където

$$
\mathbf{x} = \begin{bmatrix} x_{n+1} \\ x_{n+2} \\ \vdots \\ x_{2n} \end{bmatrix},\ \mathbf{g} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix} \text{ и } \mathbf{X} = \begin{bmatrix} x_1 & x_2 & \dots & x_n \\ x_2 & x_3 & \dots & x_{n+1} \\ \vdots & \vdots & & \vdots \\ x_n & x_{n+1} & \dots & x_{2n} \end{bmatrix}.
$$

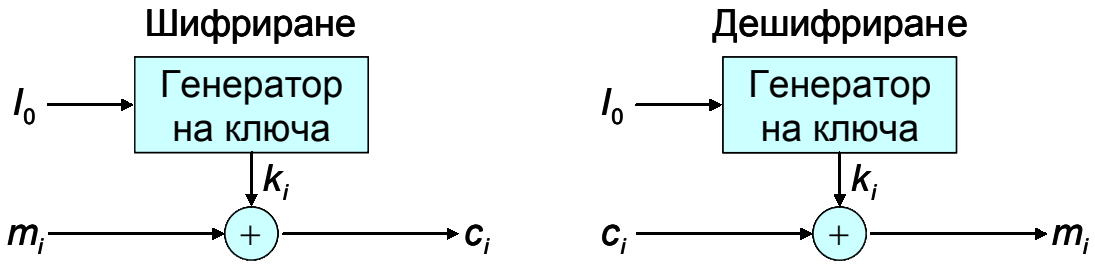
Може да се покаже, че стълбовете на $\mathbf{X}$ са линейно независими, т.е. матрицата $\mathbf{X}$ е *неизродена* (детерминантата й е различна от нула) и ще има само една обратна матрица $\mathbf{X}^{-1}$. Следователно, векторът-стълб, съдържащ стойностите на g_i, ще се определя чрез зависимостта:

(6) $\mathbf{g} = \mathbf{X}^{-1}\mathbf{x}$.

Обръщането на матрицата изисква от порядъка на n^3 операции и лесно ще се изпълнява на компютър за всяка разумна стойност на n. Например, ако $n = 100$, то $n^3 = 10^6$ и на компютър, работещ със скорост една операция за 1 микросекунда, ще му трябва 1 секунда за обръщането на матрицата. Слабото място на линейния преместващ регистър с обратна връзка се обуславя от линейността на уравненията (6), поради което се предпочита използването на *нелинейна обратна връзка* в регистъра, а това, от своя страна, прави задачата на криптоаналитика доста по-сложна, понякога дори невъзможна за решаване.

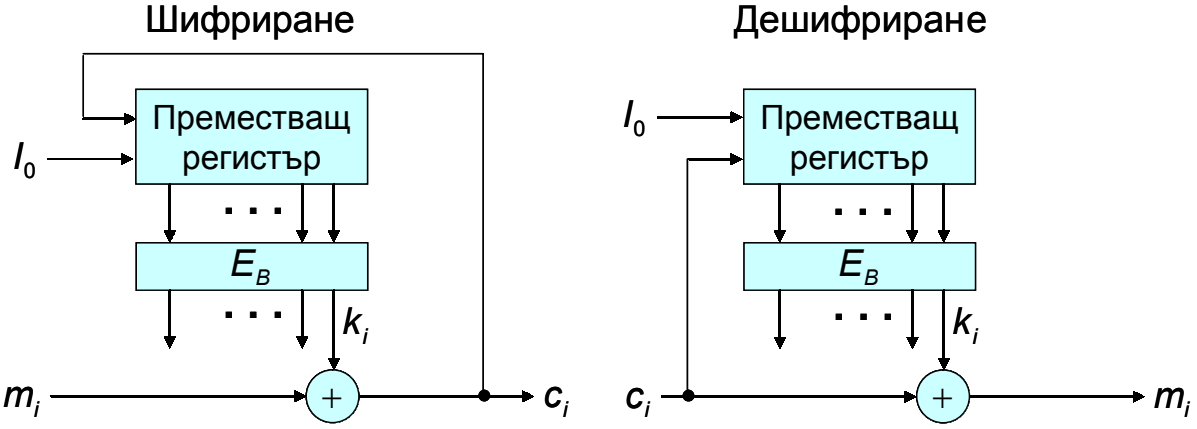
СИНХРОННИ И САМОСИНХРОНИЗИРАЩИ СЕ КРИПТОСИСТЕМИ С ПОТОЧНО ШИФРИРАНЕ

Системите за поточно шифриране могат да се разделят на *синхронни* и *самосинхронизиращи се.* При *синхронните системи* (фиг.4) ключовият поток се генерира независимо от съобщението и при загуба на символ по време на предаването задължително се изисква повторна синхронизация на генераторите на ключ в предавателя и в приемника. Началното състояние на генераторите на ключ се инициализира с помощта на известен вход I_0. Шифрираният текст се получава чрез събиране по модул 2 на i-тия символ на ключа k_i и i-тия символ на съобщението m_i. С други думи, шифрирането на символа не се разпространява по дължината на някакъв блок на съобщението. По тази причина синхронните поточни шифри не притежават *натрупване на грешки*.



Фиг.4. Синхронен поточен шифър

При *самосинхронизиращите се поточни шифри* всеки ключов символ се определя от фиксиран брой n предшестващи символа на шифрирания текст (оттук и наименованието *обратна връзка по шифър*). В такава система, ако символ от шифрирания текст се загуби по време на предаването, грешката се натрупва за n символа, но след получаване на n верни символа на шифрирания текст, системата се възстановява. На фиг.5 е показан преместващ регистър на генератор на ключ, работещ в режим на обратна връзка по шифър. Всеки изходен символ на шифрирания текст c_i (получен чрез събиране по модул 2 на символа на ключа k_i и символа на съобщението m_i) се подава обратно на входа на преместващия регистър. И тук инициализацията става с помощта на известен вход I_0. При всяка итерация изходът на преместващия регистър се използва като вход на нелинейния блоков алгоритъм за шифриране E_B. Символът на младшия разряд на изхода E_B става следващ символ на ключа k_{i+1}, който се използва при шифрирането на следващия символ на съобщението m_{i+1}. Тъй като след първите няколко итерации входът на алгоритъма зависи само от шифрирания текст, системата се явява самосинхронизираща се.



Фиг.5. Шифриране в режим на обратна връзка

ЗАКЛЮЧЕНИЕ

В статията са разгледани процесите на шифриране и дешифриране в една комуникационна система и по-специално системите с поточно шифриране, в които най-често се използват преместващи регистри с линейна обратна връзка за генериране на псевдослучайна ключова последователност. Тези схеми са твърде уязвими към *атака при известен открит текст*, поради което се предпочита реализирането на *нелинейна обратна връзка* в регистъра, а това прави задачата на криптоаналитика доста по-сложна, понякога дори невъзможна за решаване.

ЛИТЕРАТУРА

[1] Антонов, П., С. Малчев. Криптография в компютърните комуникации. Варна, 2000.

[2] Скляр, Б. Цифровая связь. Теоретические основы и практическое применение. Москва, Вильямс, 2003.