

**08. 04. 2008 г.**

## **Приложна криптография (ПК)**

Криптографията е математическа наука и изучава методите, способите, средствата за преобразуване на данни с цел:

- да се скрие семантичното значение на данните;
- да се предотврати неправомерното им използване;
- да предотврати откриването на евентуални, случайни или преднамерени изменения в данните.

### **Основни понятия.**

**Криптиране (шифриране)** – процес на преобразуване на данни (открит текст – о.т.), чрез специални криптографски трансформации, в резултата, на което се получава т.н. криптиран или шифриран текст.

**Декриптиране (дешифриране)** – обратен криптографски процес. Процес на преобразуване на шифриран текст в явен текст или в открит текст с използване на т.н. криптографски ключ, за получаване на открития текст.

**Открит текст** – документ, който трябва да се криптира. Елементи – букви, срички, символи, цифри, думи, абзаци, изречения.

**Криптиран текст** – получен в резултат на шифрирането. Съдържанието на данните в този текст е нечитаемо без използване на криптографски средства. Елементи – елементи от шифъра, които се използват като заместители от открития текст.

**Криптографски алгоритъм** – за функционално преобразуване на открития текст, разбираема последователност от символи и съответно тяхното правилно възстановяване.

**Криптографски протокол** – съвкупност от правила за използване на криптографски алгоритъм и правила за обмен на данни.

**Криптографска система (крипто система)** – съвкупност от два или повече криптографски алгоритма съответно криптографски протоколи.

**Криптографски ключ** – множество от символи и/или числа, което се използва за криптиране/декриптиране на данните. Чрез него се управляват

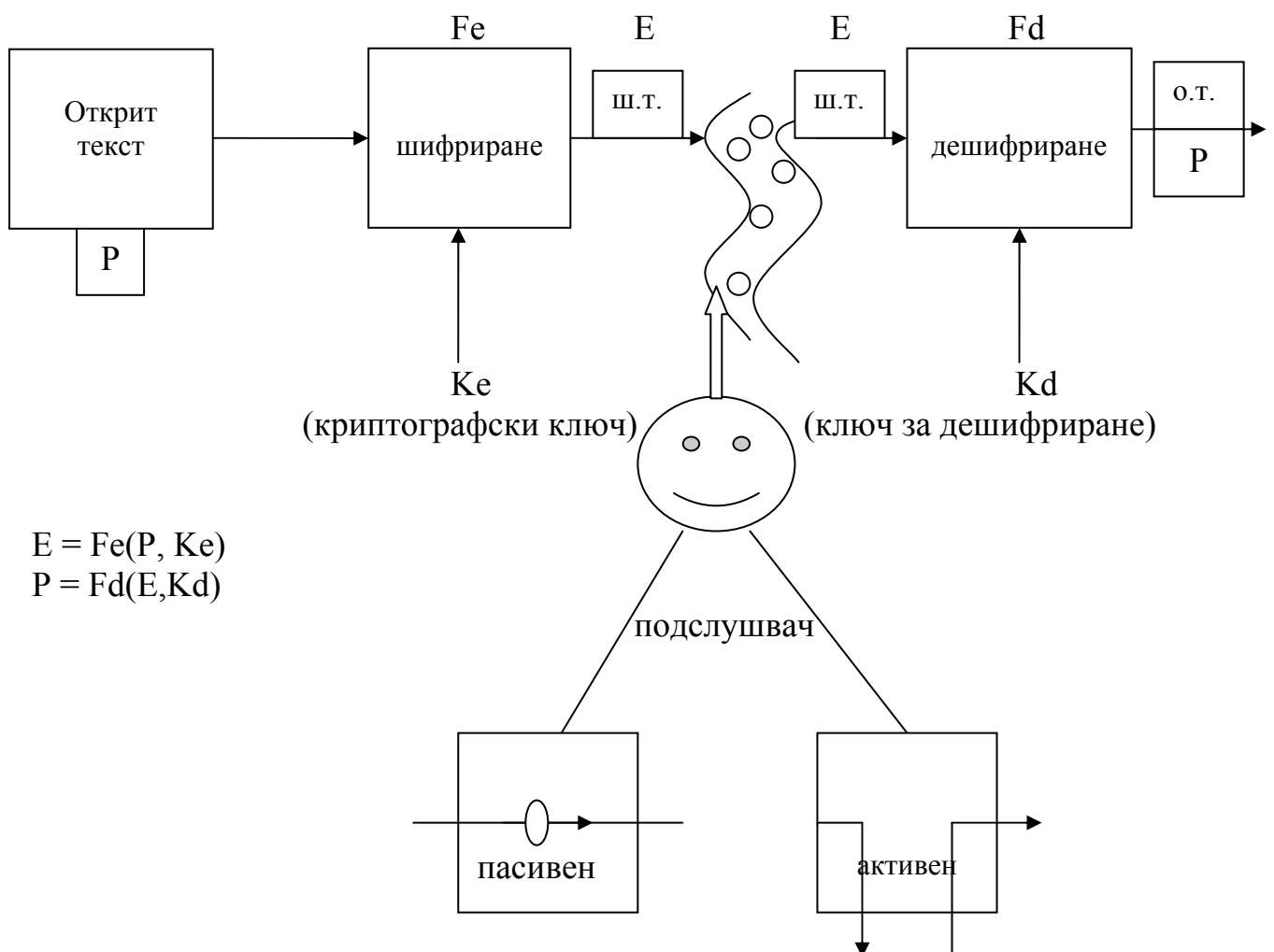
процедурите по криптиране/декриптиране на данните. Има две групи криптографски ключове:

- секретни – ако изискванията на крипто системата са да бъде запазен в тайна;
- несекретни – ако изискванията на крипто системата налагат той да бъде публично оповестен и достъпен.

**Криптографско средство** – техническо средство (апаратура, хардуер) и/или програмно средство използвани за криптиране/декриптиране на данни и/или за генериране и тестване на криптографски ключове.

**Криптографска защита/криптографска сигурност** – прилагане на криптографски методи и средства за защита на данните при тяхното съхранение и обмен (защита срещу разкриване от неоторизирани лица, неправомерно ползване и неправомерна промяна).

**Криптографска мрежа** – съвкупност от съвместими криптографски средства с общо администриране на криптографски ключове осигуряващи криптографска защита на съхраняваната или обменяна информация.



Процеса на получаване на открити данни от шифриран текст в общия случай без наличието на криптографско средство и ключ се нарича криптоанализ.

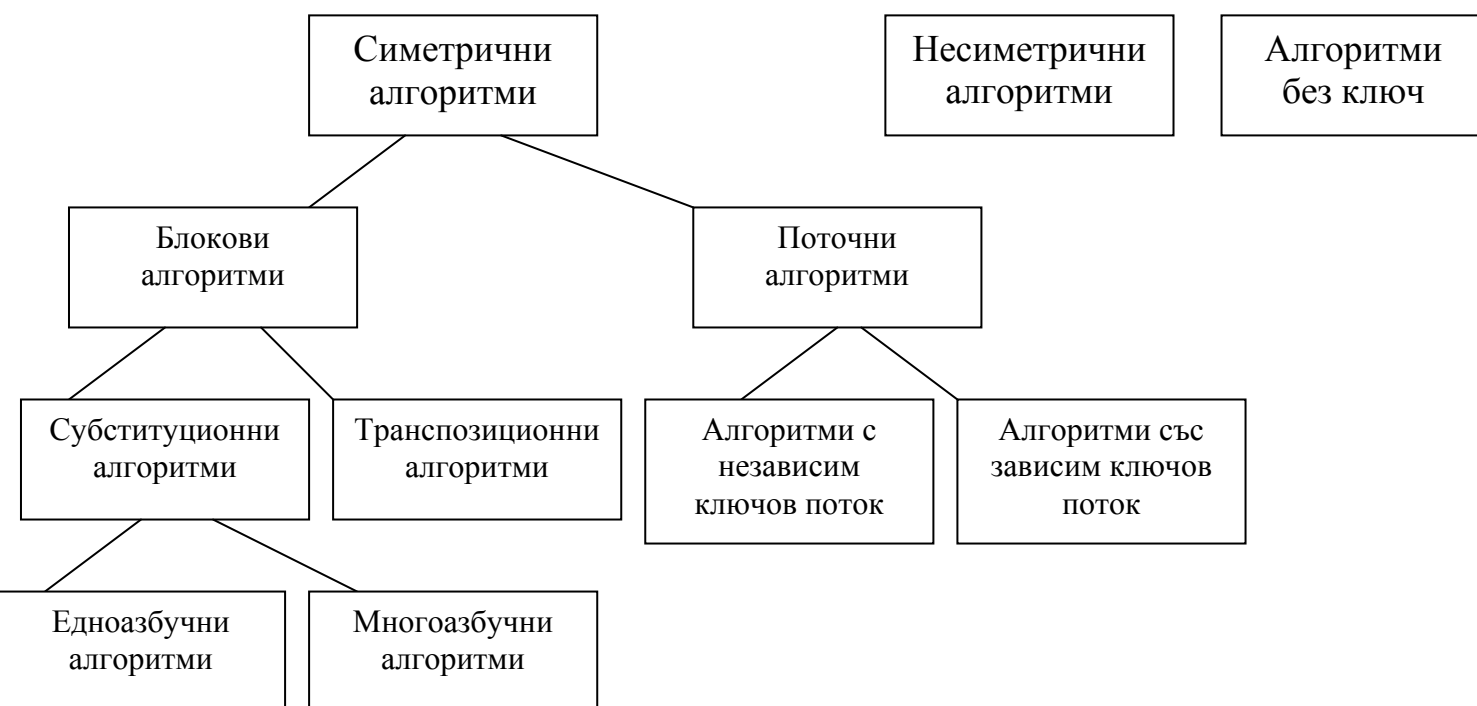
Криптоанализа е наука занимаваща се с разработване на методи и средства за разкриване на тайната на криптографските системи и за оценка на сигурността им.

Криптографията и криптоанализа са двете страни на науката криптология. Две взаимно свързани страни. Специалистите занимаващи се с тези науки са криптографи – криптография и криптоаналитици – криптоанализ.

### Класификация на криптографските алгоритми (шифър)

Криптографския алгоритъм се базира на функциите шифриране и дешифриране на данните. При шифриране се използва ключ за шифриране, който е известен само на шифриращата страна или е всеобщо достъпен. Ключът за дешифриране може да е същия, с който е извършено шифрирането или за дешифриране да се използва друг ключ известен само на приемащата страна.

Всяка една криптографска система се състои от два или повече базови криптографски алгоритъма и се характеризира с избраните режими за тяхната реализация. Съответния базов криптографски алгоритъм и реализираните режими подлежат на класификация по различни класификационни признаци. Като класификационен признак на първо ниво най-често се използва свойството симетрия на алгоритми. По този признак базовите криптографски алгоритми се разделят на симетрични и несиметрични.



При симетричните и несиметричните криптографски алгоритми се прибавят и алгоритмите без ключ. Те са разделени на хеш алгоритми (hash – хеширане) и генератори на ключов поток или на случайна последователност.

### Същност на криптирането

При симетричните криптографски алгоритми процеса на шифриране и дешифриране се извършва с използване на един и същи криптографски ключ.

$$K_e = K_d = K_s$$

Те се наричат още алгоритми със секретен ключ ( $K_s$ ).

**Субституционните блокови алгоритми** – от субституция - това е математическа операция извършена еднозначно и обратимо съпоставяне на символи в открития текст с други символи в шифрирания текст или блок символи в отворения текст се съпоставят на друг блок символи в шифрирания текст.

Субституционен блоков алгоритъм – още се нарича “Шифър на Цезар”.

А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ю Я  
 ы ю я а б в г д е ж з и й к л м н о п р с т у ф х ц ч ш щ ъ

п о т р е б и т е л  
 м л п н в ю е п в и

Субституционните блокови алгоритми още се наричат заместителни шифри или алгоритми.

**Транспозиционните блокови алгоритми/шифри** – това е математическа операция, която се състои в еднозначно и обратимо преобразуване на блок от краен брой символи, чрез пренареждане (пермутиране) по определена схема с определен ключ.

	1	2	3	4	5
1	с	ъ	о	б	щ
2	е	н	и	е	с
3	ъ	о	б	щ	е
4	н	и	е	с	ъ

Съобщение – ключ 42135  
 бещсъноисеъноибещсеъ – криптографски ключ

Символите от отворения текст се запазват, но само се разменят.

Транспозиционните блокови алгоритми още се наричат разместителни шифри или алгоритми.

### Основни принципи на криптографията

Основните принципа са два. Първия принцип осигурява секретност на криптографските ключове. А втория осигурява и разработва методи, способности и средства за обмяна на криптографските ключове. На базата на тези принципи са разработени и пет постолата:

- първи постолат – сигурността на доверената криптографска система не зависи от секретността на използваните криптографски алгоритми, а зависи основно от опозване на криптографския ключ;
- втори постолат – сигурната, силна криптографска система притежава голямо пространство на криптографските ключове;
- трети постолат – сигурната, силна криптографска система произвежда шифриран текст, който притежава напълно случайно статистическо разпределение на символите в него, и който шифриран текст не подлежи на статистически анализ;
- четвърти постолат – цената на “разбиване” криптографския алгоритъм, трябва да бъде много по висока от цената на защитаваната информация;
- пети постолат – времето за “разбиване” на криптографския алгоритъм трябва да бъде по дълго от времето, за което защитаваната информация запазва своята актуалност и представлява интерес. За постигане на устойчивост или строгост на криптиращия алгоритъм съществено значение играят криптиращите ключове и техните дължини.

Броят на единиците и нулите определя дължината на криптографския ключ. Дължината на криптографския ключ се определя така че да е защитен от пълно изброяване – комбиниране (атака на грубата сила).

Дължина на ключа	Пространство на ключовете в двуйчна бройна система	Пространство на ключовете в десетична бройна система	Криптоанализ 10 на 20-а 1 ключ 1Ms	10 на 6-а/1Ms
32 бита	2 на 32-а степен	4,3x10 на 9-а степен	35,8 минути	2,15 Ms
128 бита	2 на 128-а степен	3,4x10 на 38-а степен	3,4x10 на 24-и години	5,4x10 на 18-а години

При дължина на ключа 32 бита са възможни комбинации от ключове – 2 на 32-а степен.

128 бита са равни на 2 на 128 степен.

Дължината на криптографския ключ определя т.н. горна граница на устойчивост на криптографската система или атакуващата система винаги може да използва атаката “груба сила”, като ноправи проверка с всички възможни ключове, за да намери верния.

Към момента използваните изчислителни средства и технологии са в състояние да решат в приемливо време задачата за намиране на верния брой ключове или до 10 на 20 степен варианта.

Ключ с дължина 128 бита се определя, като гаранция за създаване на алгоритъм устойчив на атаката “груба сила”.

Криптографските цели са да се избегнат усложнения от дължината на ключа. За целта се създават алгоритми, за които теоретично се доказва, че са устойчиви на най-сложните методи за крипто анализ, а именно – линейния и диференциалния.

Ако даден алгоритъм е устойчив на анализ на съвременните криптографски методи, той може да бъде предложен за развитие.

Създаването на сложни алгоритми са резултат на работата на криптографите и криптоаналитиците.

### **Управление на криптографските ключове**

Това е информационен процес влючващ реализирането на три основни функции:

- генериране;
- разпределение;
- съхранение.

Освен понятието криптографски ключ се използват и понятията ключова информация, ключов документ и ключов материал. Съвкупността от всички действащи в системата ключове представлява ключовата информация.

**Генерирането** на криптографските ключове се реализира по определена схема за всеки алгоритъм, като се спазват изискванията на съответните стандарти. В криптографията се използват както случайни, така и псевдослучайни стойности за криптографските ключове. За да бъде един псевдослучаен ключ достатъчно надежден, той трябва да бъде неопределен или небива да се допуска прогнозиране на всеки следващ бит в поредицата, дори ако се знае детайлно алгоритъма за генериране и всички предходни битове от ключа.

Както криптографските алгоритми, така и криптографските ключове се полагат на криптографски анализ. При криптографския анализ на

криптографските ключове целта на криптоаналитика е да направи криптографския ключ устойчив на криптографски атаки. Криптографските ключове в една криптографска система са структурирани в йерархична структура – най-отгоре стой главния ключ (maste key).



Работния ключ се използва за шифриране на данни, а също така и за тези които ще бъдат обменяни. Самия криптографски ключ се предава в шифриран вид. Вторичните ключове се ползват за шифриране на работните ключове. Персоналните ключове са притежание на всеки абонат. Чрез първичните ключове се шифрират както вторични, така и цялата информация. Естествено основен е главния ключ. При негова промяна се променя и цялото съдържание на йерархията.

За да се съхрани информацията (данни с криптографски ключ излезли от употреба) е необходимо да се разработи и използва подходящ механизъм за контрол при работата на криптографските ключове.

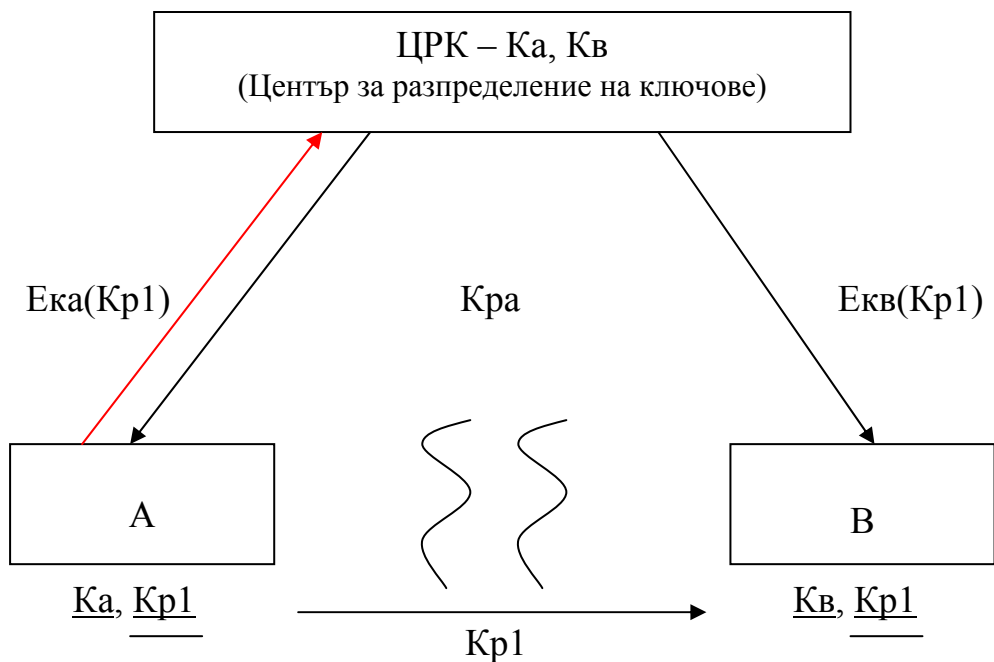
От своя страна криптографските ключове се разделят на дългосрочни и краткосрочни. Тази се нарича така на базата на т.н. криптопериод (време на съществуване и използване на криптографския ключ). Дългосрочните се използват за шифриране на различни видове криптографски ключове в йерархичната система. Краткосрочните са за данни на външен носител, за големи обеми от данни при комуникация в мрежа, при работни станции и сървари.

**Разпределението** на криптографския ключ е важен процес. Неговата ефективна реализация зависи от употребата и съхранението на криптографските ключове. Има две изисквания:

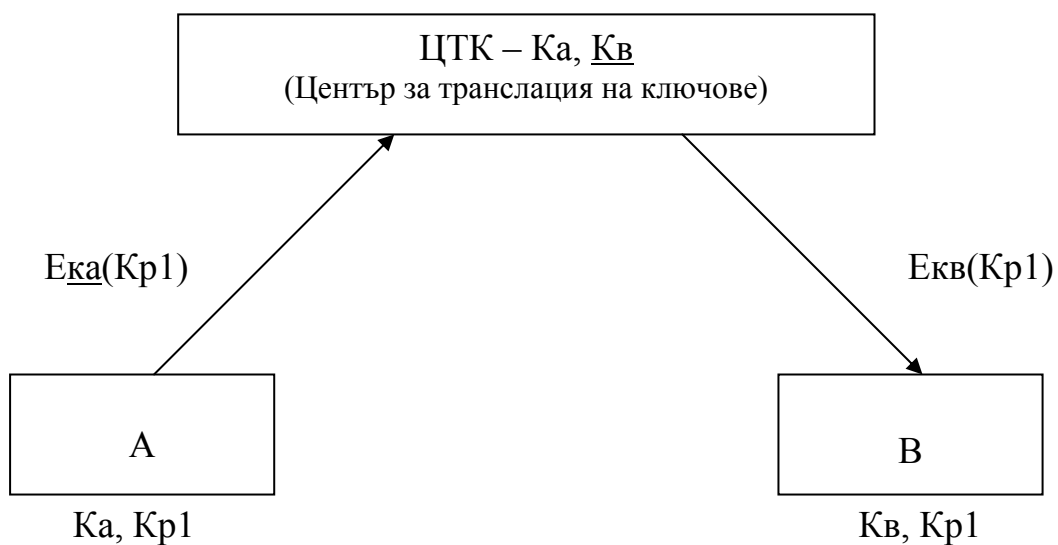
- оперативност и точност;
- запазване целостта и секретността на криптографските ключове.

Методи за разпределение:

- използване на център за разпределение на ключове;



- модел с център за трансляция на ключове;



ЦТК извършва автентификация на участниците в комуникационния процес или определяне на периода от време, в който тези ключове са легални и могат да бъдат използвани.

- модел с пряк обмен на ключове между А и В – няма гаранция, че А и В ще получат автентичния текст или че те са тези между които се осъществява комуникацията.

**Съхранение** на ключовете – извършва се по три начина:

- в специални криптографски устройства (транспортен модул) с памет защитена от несанкциониран достъп;
- използване на външна памет за съхранение на криптографския ключ (смарт карта, флаш памет);
- съхранение на криптографския ключ в шифриран вид в персоналния компютър.

### **Симетрични криптографски алгоритми.** **Блоково шифриране**

Блоковото шифриране се извършва с апаратни и блокови средства. Може да се реализира висока скорост на шифриране и дешифриране.

Блоковото шифриране е преобразуване на символен низ с фиксирана дължина (открит текст) в изходен блок с фиксирана дължина (шифриран текст) с помощта на секретен криптографски ключ. Фиксирания блок най-често е с размери 64, 128 бита. При блоково шифриране криптирания ключ е фиксиран и се нарича блок шифър. Размера се определя от съображенията за сила на криптографския алгоритъм и трябва да бъде достатъчно голям, за да осуетява прости атаки срещу съобщенията, като тип “пълна атака”.

криптоаналитик	C1	C2	C:	Cj		Cn
ключове						
K1 K2 Ki Km	P21			P:i		Pnm

Това е специален речник на криптоаналитика с явните и криптирани съобщения. А също така е и речник на блоковете.

**Анализ на честотата на блокове** – при този тип атаки се използват статистически методи за определяне на възможност на повторно провеждане на даден тип блок. Колкото е по голям блока, толкова е по малка вероятността за повторно провеждане във времеви блок.

**Аналитична атака** – при този тип устойчивостта е толкова по-голяма, колкото е по-голяма символната зависимост между открития текст и шфрирания текст.

Да се осигури по-голяма степен на сигурност в блоковите шифри трябва да се използват по-голям брой криптографски ключове, като се допуска и повторно използване на някои от тях. Поради тази причина блоковото шифриране и дешифриране се използва в определен брой цикли с различен брой ключове по-малки като размер от осовния брой ключове. Изпълнението на отделните цикли в криптографските алгоритми се извършва от множество криптографски ключове.

Кр:Кр1+Кр2.....Крm

Извършва се обработка само с цял блок с фиксирана дължина, за това се разрешава само случая когато размера на съобщението не е кратен на размера на блока.

512:64=8 – стандартен блок

560:64=8+48 – къс блок

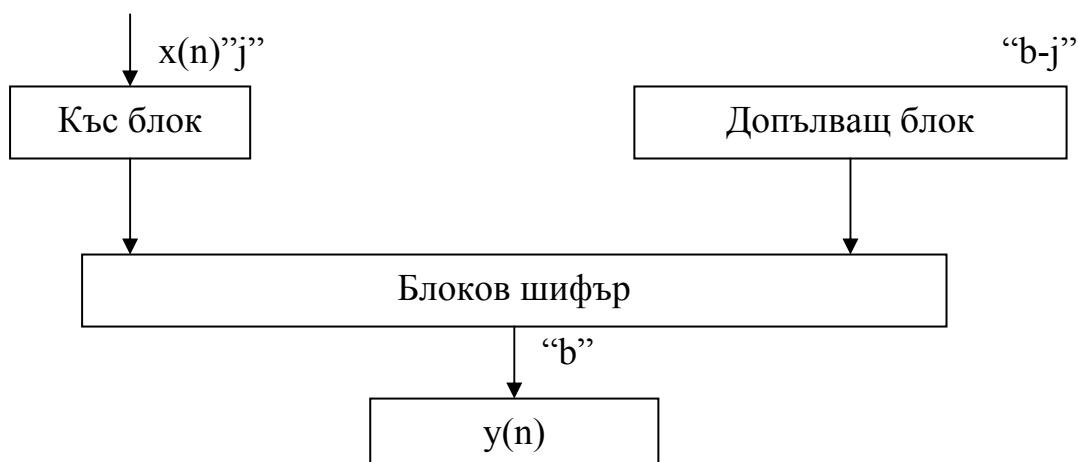
P: x(1) x(2) x(3) x(4) ..... x(i) ..... x(n)

(открит текст)

Всички до x(n) са стандартни блокове. X(n) е къс, но може да бъде и стандартен.

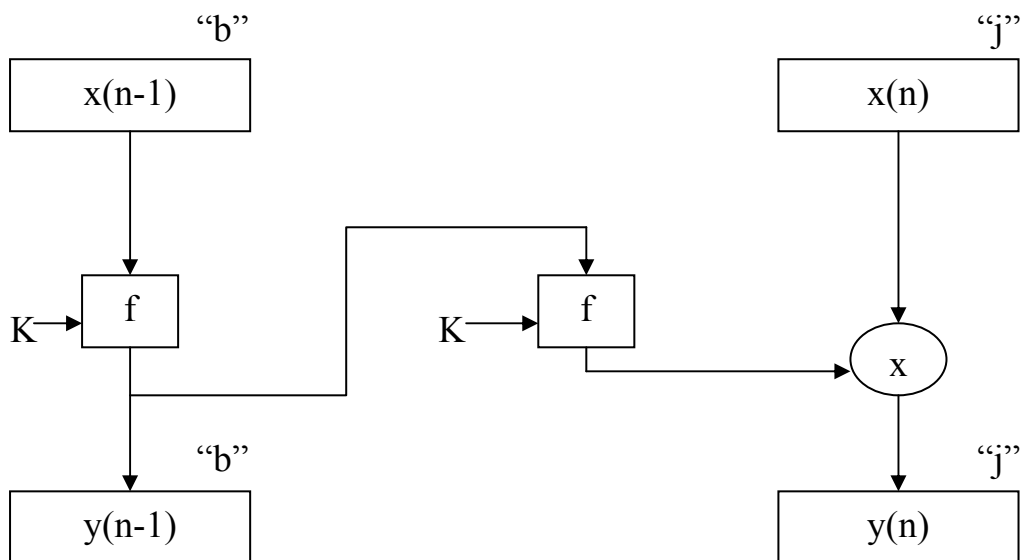
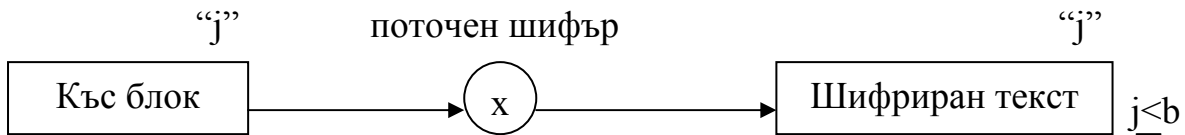
**Допълване на къс блок да стандартен блок:**

- първи метод – реализира се като се използват допълващи битове получени от случаен или псевдослучаен процес;

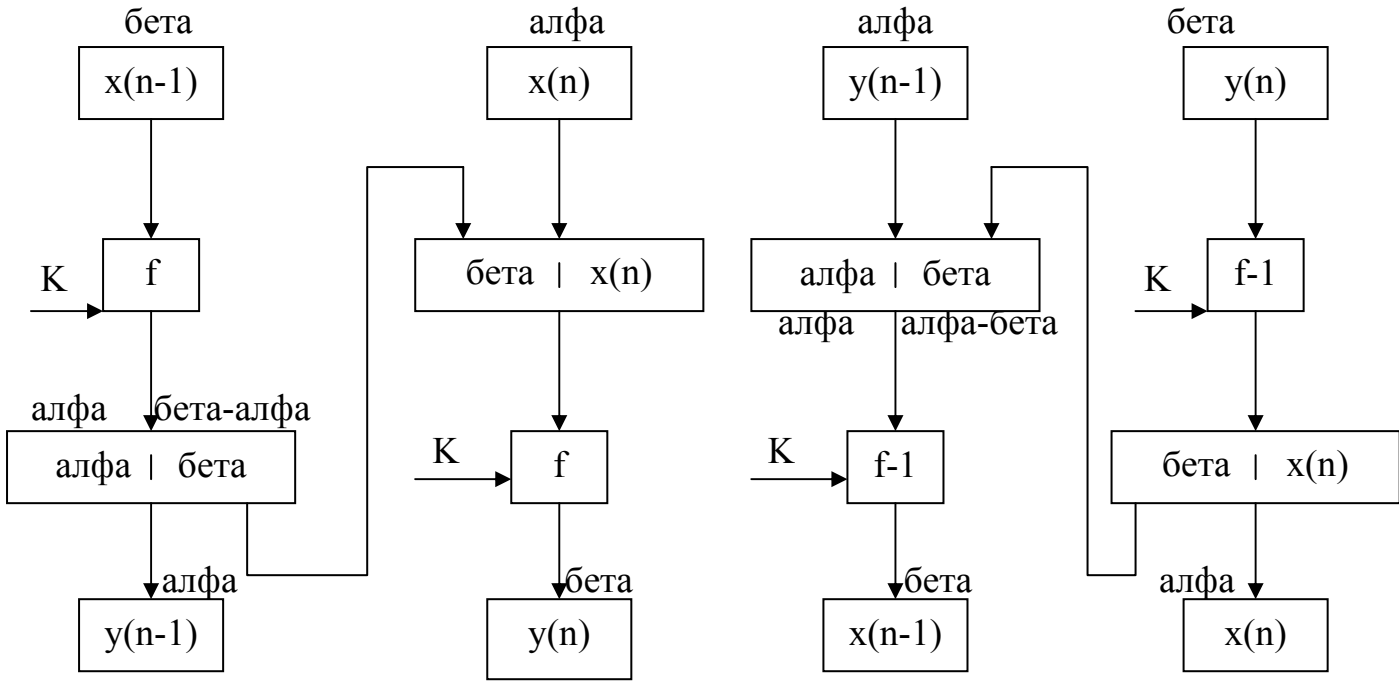


Този метод се използва когато се предават съобщения с променлива дължина. Така се увеличава размера на предаваните данни.

- втори метод – не е в допълване на късия блок, а късия блок се шифрира с поточен шифър. Стандартния блок се шифрира с блоков шифър;



- трети метод – чрез вътрешно прехвърляне на символи от стандартен блок към къс блок, от едната процедура към другата.



шифриране

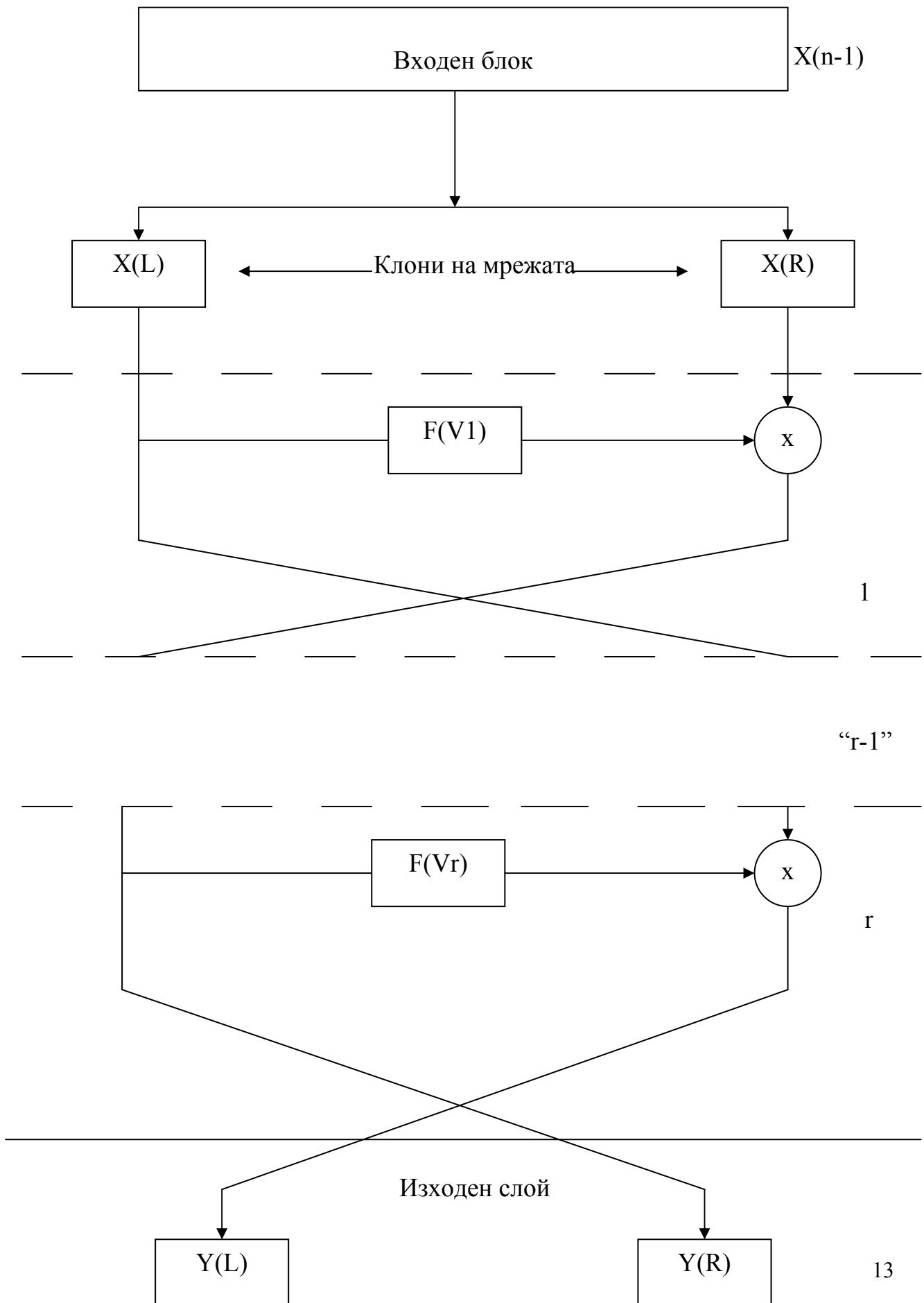
дешифриране

1011111010

101101001 - ключ

# Класическа схема (мрежа) на Feistel (Фейстел).

Всички известни блокови схеми почиват на тази схема.



Този метод осигурява такава обработка на входния текст, при която междинни резултати на шифриране на една част се наслагват върху други резултати от същия блок.

V1, VR – параметри на мрежата на Фейстел, изчислени на базата на ключ. Действие, което включва изчисление на функцията (F) последващо слагане на получения резултат в друг клон на мрежата и смяна на местата на двата клона се нарича цикъл в мрежата на Фейстел.

Общия брой на цикъла е R, варира между 8 и 32 бита. По високия брой води до устойчивост, но изисква повече време.

Разликата между шифрирането и дешифрирането се състои в използване на различни ключове.

FEAL – Fast Encipherment Algorithm – създаден в Япония 1989 г., патентован в много страни. Характеристики: - реализира класическата мрежа на Фейстел – 64 бита, два клона – лив и десен, с различен брой цикли. Характеризира се с простота на функцията в сравнение с останалите алгоритми. Неустойчив на линейния и диференциалния анализ.

SAFER – Secure and Fast Encryption Runtime – Cylink Co. 128 бита.

Характеризира се с това че разделя мрежата на Фейстел на четири клона и операцията за шифриране се различава от тази за дешифриране. Устойчив на известните атаки, а също и на линейния и диференциалния анализ.

CAST – Carlisle Adams Sttattout Taver (PGP).

Характеризира се с класическа мрежа на Фейстел, два клона, устойчив, 128 бита.

RS -5 – 1995 г. от RSA – Data Security Inc.

Характеризира се с променливи цикли – 16, 32, 64, както и дължина.

RS/5 – w/r/b = AES, който заменя DES.

**10. 04. 2008 г.**

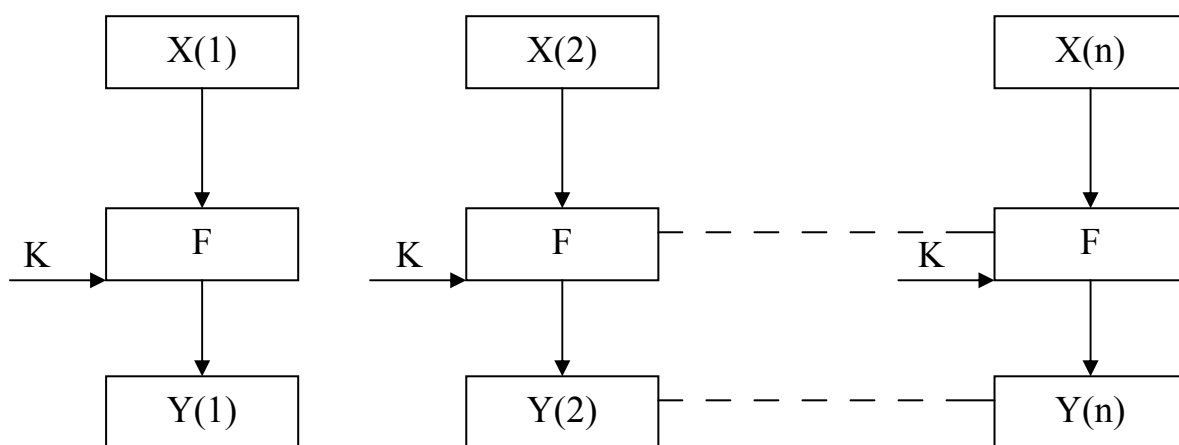
**Режими на работа на блоковите алгоритми**

Основното предназначение на различните режими е да повишат устойчивостта на криптографските алгоритми спрямо известните атаки за провеждане на криптоанализ. Съществуват четири режима:

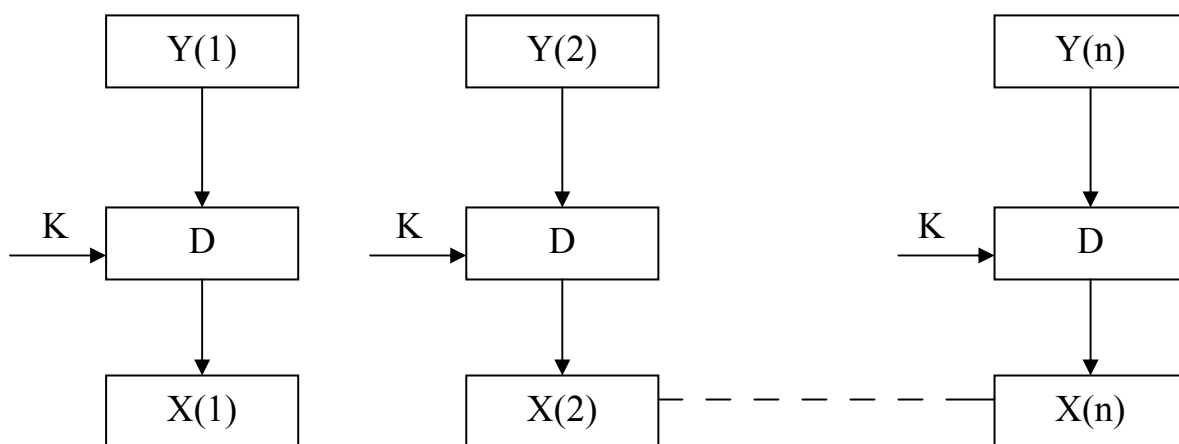
- **режим на проста замяна** – при него всеки блок от открития текст се шифрира независимо от останалите;

P: X(1), X(2),.....X

$Y1(k)=F[X(1)]$



Шифриране



Дешифриране

Electronic Codebook Mode – ECB – режим на електронна кодова книга.

Името произлиза от възможността да се създаде електронна книга за всеки криптографски ключ, в която да се съхраняват записите за всеки шифриран блок и съответстващия му блок с шифриран текст.

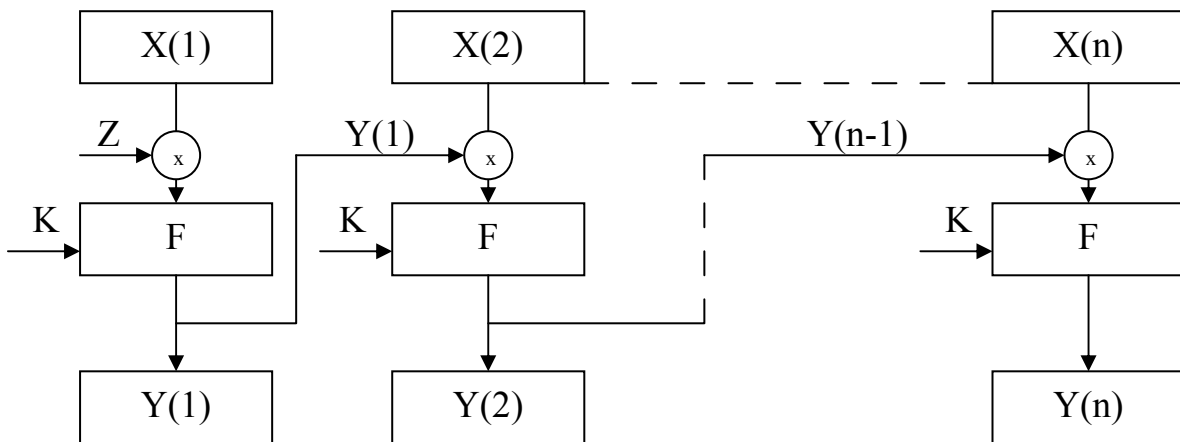
2 на степен 64 – фиксирана дължина на блока, т.е. за един ключ.

Еднаквите входни блокове се трансформират с еднакви изходни блокове с шифриран текст, за това той е неустойчив срещу атаки, като най често срещаната атака е с речник с блокове. Това ограничава приложението на този режим.

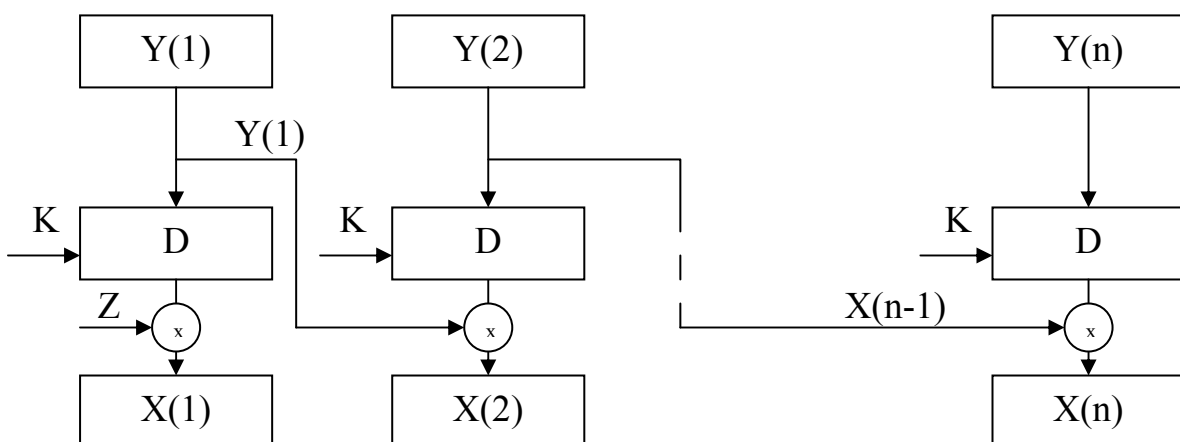
**- свързано блокова шифриране;**

P: X(1), X(2),.....X(n)

E: Y(1), Y(2),.....Y(n)



Шифриране



Дешифриране

Z – начален (инициализиращ вектор);

Y(n) зависи от началния текст и Z, т.е. се осигурява висока междусимволна зависимост.

- **режим на обратна връзка;**
- **режим на обратна връзка с шифриран текст.**

Във всички блокови алгоритми до 2000 г. има реализация на Фейстел и тези режими. → FEAL; SAFER; CAST; RC-5.

DES – реализиран като 56 бита; ключа е с 64 бита; 56 бита за шифриране и 8 за начален вектор.

3DES – Triple DES – извършва три реализации.

Новия стандарт на САЩ, а и на света се нарича Advanced Encryption Standart – AES.

AES – 02. 09. 1997 г. – американския институт NIST обявява конкурс за нов стандарт. За разработването му са поканени най-известните криптографи, като се работи по стандарт за ползване през следващите 15 години. Алгоритъма трябва да е симетричен, да е блоков, да обработва блокове с дължина 128 бита, да работи с три варианта ключове – 128, 192, 256 бита; да не се усложнява допълнително структурата на алгоритма. Целта е всички заинтересовани страни да могат да проведът самостоятелно изследване, за да се уверят, че алгоритъма работи коректно.

През първия етап на конкурса има 15 предложения обработвани 2 години, като в резултат се отсяват 5 алгоритъма допуснати до втори кръг:

- първи алгоритъм – MARS (IBM) – на базата на класическата схема на Фейстел и множество математически задачи за обработване на данни. Всеки е естван за бързина и скорост на обработка на данните. Скоростта му е 8 Mb/s (мегабита на секунда).
- RS-6 (RS-5) – скорост 12 Mb/s;
- Serpent – от Израел, Великобритания и Норвегия – с най-слаба скорост - 2 Mb/s;
- Twofish – от американски изследователи - 11 Mb/s;
- Rijhdael - 7 Mb/s – той единствен не се базира на мрежата на Фейстел. Определя се като нетрадиционен блоков алгоритъм.

Освен за скорост алгоритмите са тествани и за устойчивост.

На 02. 11. 2000 г. е обявен последния алгоритъм за победител Rijhdael или AES.

Той е удобен за апаратна и програмна реализация на различни процесори. Може и паралилно да извършва операции, с което се повишава скоростта на шифриране и дешифриране. Броя на участниците в криптографската мрежа е повече от два. Ключовете също нарастват.

$$= N \Rightarrow \frac{N(N-1)}{2}$$

A●  
Kab  
Kap

B●  
Kab  
Kap

P●  
Kap  
Kbp

Ключовете трябва да бъдат сменяни през определен период от време (32, 56, 64, 129 бита и т.н.).

### **Симетрични криптографски алгоритми. Поточно шифриране.**

При поточните алгоритми всеки символ от открития текст се шифрира и дешифрира независимо от останалите или преобразуването не е функция на една и съща криптографска трансформация. Следователно се налага създаване на секретен ключ, като последователност от символи, с която се реализира алгоритъма. За целта се използват букви, цифри и други разрешени символи, при което отворения текст се преобразува в съответствие на криптографския ключ. Размера на криптографския ключ трябва да е по голям или равен на размера на открития текст.

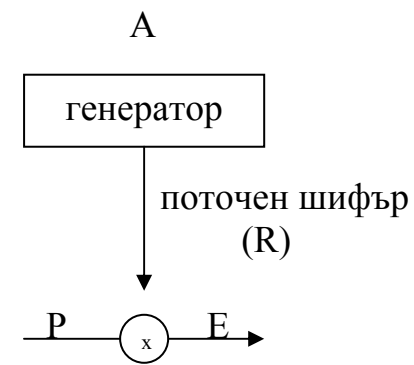
Една и съща гама не би следвало да се използва повторно във времето и в пространството, тъй като това би довело до значително снижаване в устойчивостта на алгоритъма.

Алгоритъма трябва да бъде устойчив на атаки установяващи се на статистически анализ.

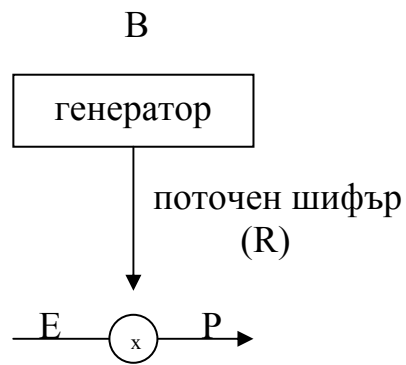
Принципа е използване на основен криптографски ключ за получаване на различен брой гама за гамиране. Най сигурен способ за постигане на сигурност е честа смяна на ключа.

Вернам – 1926 г. – метод на Вернам.

Създадени от тогава до сега поточните шифри се оформят като самостоятелна група поточни криптографски алгоритми.



$$E(i) = P(i) \otimes R(i)$$

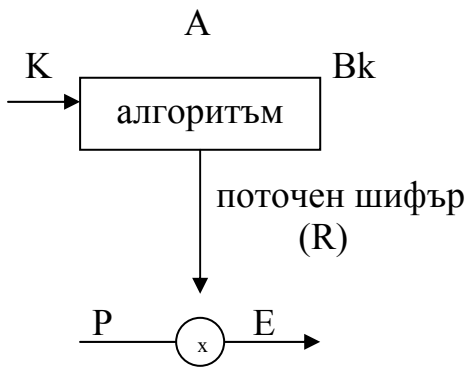


$$P(i) = E(i) \otimes R(i)$$

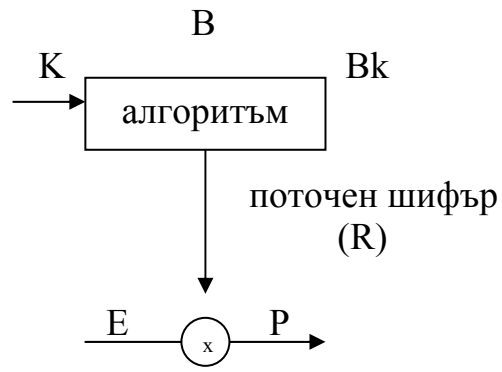
Случайна последователност от битове генерирани от генератор на случайни числа аналогично на блоковия шифър, но не е с фиксирана дължина.

Обработката на шифриран текст с поточен шифър води до появата на открит текст.

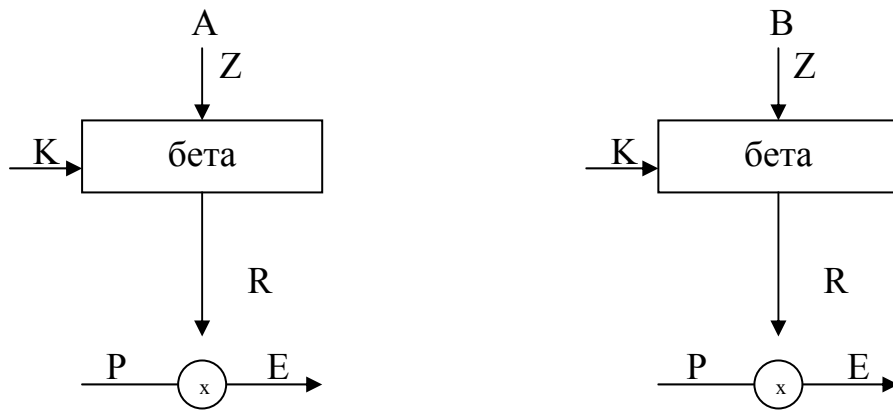
Генератор на случайни числа се реализира като алгоритъм, който да се реализира както при подателя, така и при получателя.



$$B_k \{B_{k1}, B_{k2}, \dots, B_{kn}\}$$



Почти всички поточни шифри използват умножение по вектор.

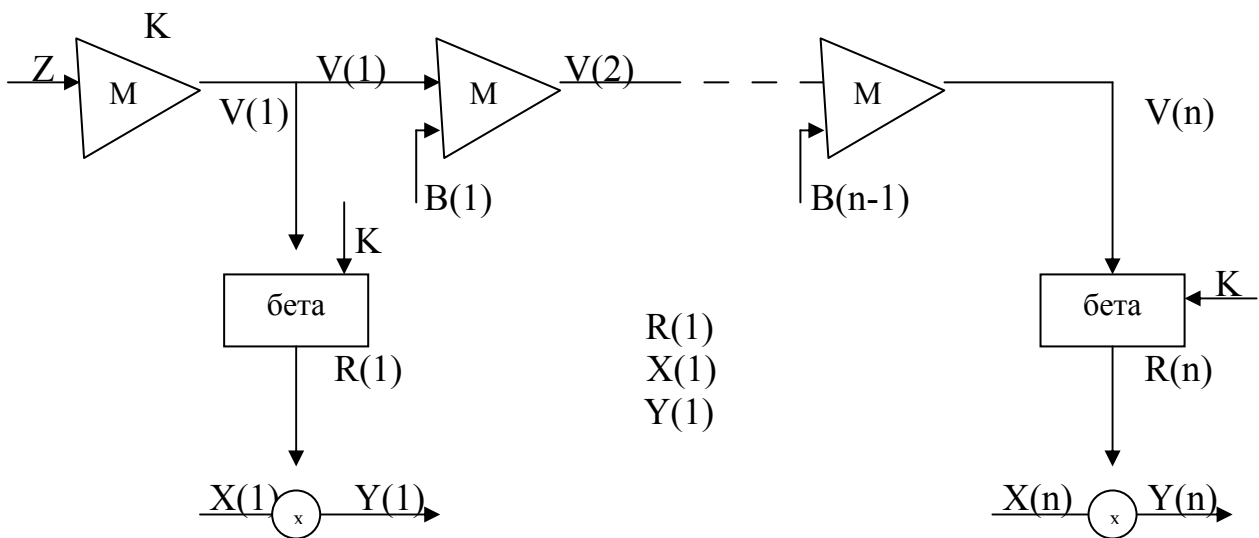


Различните стойности на  $Z$  при използване на един и същи криптографски ключ генерират различни поточни шифри.

$Z$  – случайна или псевдослучайна стойност, при която не се допуска повторяемост в определен период от време. Стойността му се взема от таймера на компютъра.

За разлика от секретните криптографски ключове инициализиращия вектор  $Z$  може да бъде и явен или несекретен параметър. За двете страни  $Z$  трябва да е еднакъв. Възможно е получателя да генерира вектор и да го изпрати на подателя.

### Стъпаловиден поточен шифър



$$V(1) = M(z)$$

## Съпаловиден поточен шифър с обратна връзка

Явен текст      100101101011 }  
Поточен шифър 110011001010 } XOR

Две последователности от записани цифри.

Ако поточния шифър е безкраен и безмислен на принципа на безмислено връщане, в този случай теоретично и практически шифъра е нерзчитаем.

След втората световна война американското правителство разрешава на Клод Шенън да публикува теория за информацията в секретните системи и математическа теория на информацията. Така се появяват и първите публикации по проблемите на криптографията.

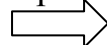
Поточните криптографски алгоритми имат по ограничено проложение в сравнение с блоковите.

## Несиметрични (асиметрични) криптографски алгоритми

Идеята за асиметрични криптографски алгоритми е разработена през 1976 г. независимо от два екипа – Дифи и Хелмън и Меркеле.

Най-съществен принос имат **R**ivest, **S**hamir, **A**deemen – RSA – стандарт за асиметрична криптография. Тяхната фирма е водеща в разработване на криптографски средства на базата на асиметрични криптографски алгоритми.

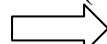
Всяка мрежа разполага с два ключа – публичен ( $K_p$ ), или още общодостъпен, открит, общоизвестен, явен и секретен ( $K_r$ ), или още таен, личен, частен.

Криптографска система с публичен ключ се нарича публична криптография,  асиметрични криптографски алгоритми.

От изчислителна гледна точка е много трудно да се определи секретния ключ, ако се знае само публичния ключ.

Криптографски алгоритми с публичен ключ се базират на шифриращи функции характеризиращи се със свойството еднопосочност.

$Y=F(x)$

 Изчислява се лесно, но притежава обратна функция, която се изчислява изключително трудно.

Преимствата и внедряването на публичната криптография води до съществено намаляване на броя на необходими криптографски ключове.

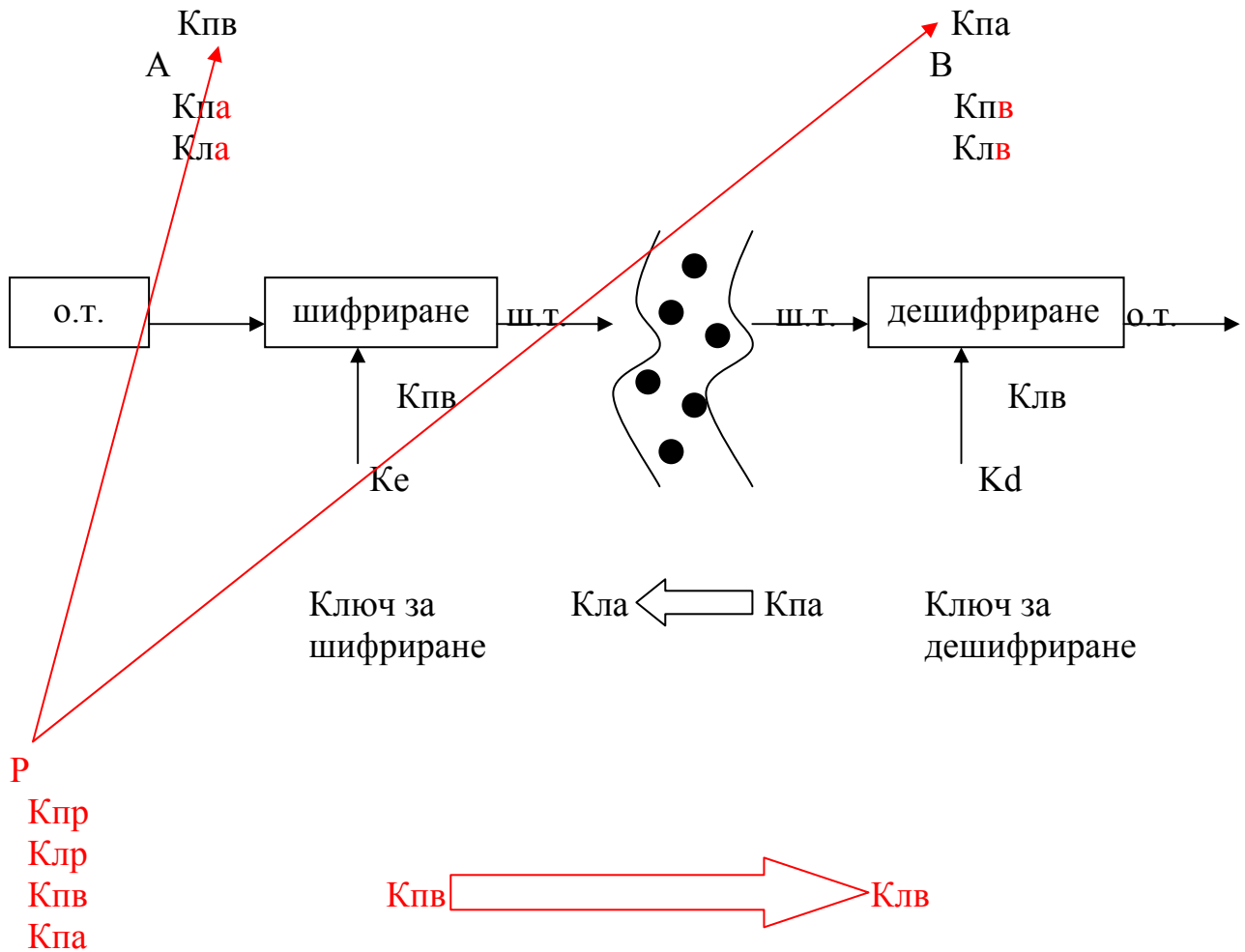
$N(N-1)/2$

Ако има  $N$  участници, те притежават двойка ключове, т.е.  $N$  на брой ключове.

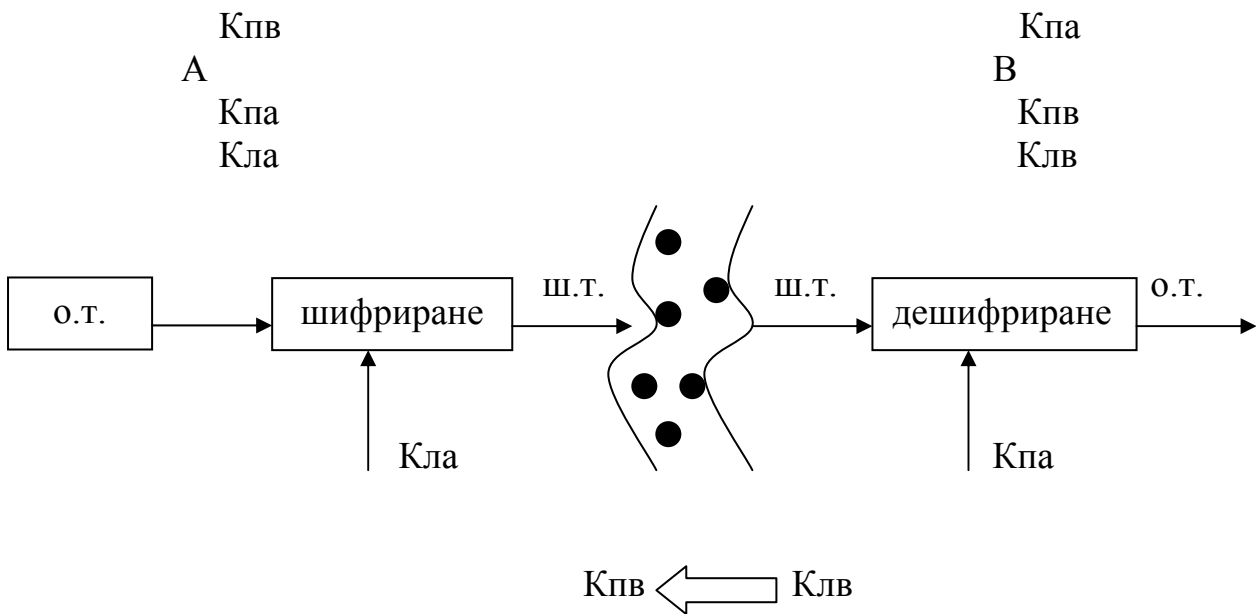
Няма проблем за обмена на криптографските ключове, т.е. не е необходимо да се търси втори канал за предаване на ключовете.

Публичната криптография предоставя по голям брой услуги по сигурността в съвременните информационни системи в сравнение със симетричните криптографски алгоритми.

**Криптиране с използване на публична криптография**  
**Криптиране**



## Цифрово подписване ( Digital signature)

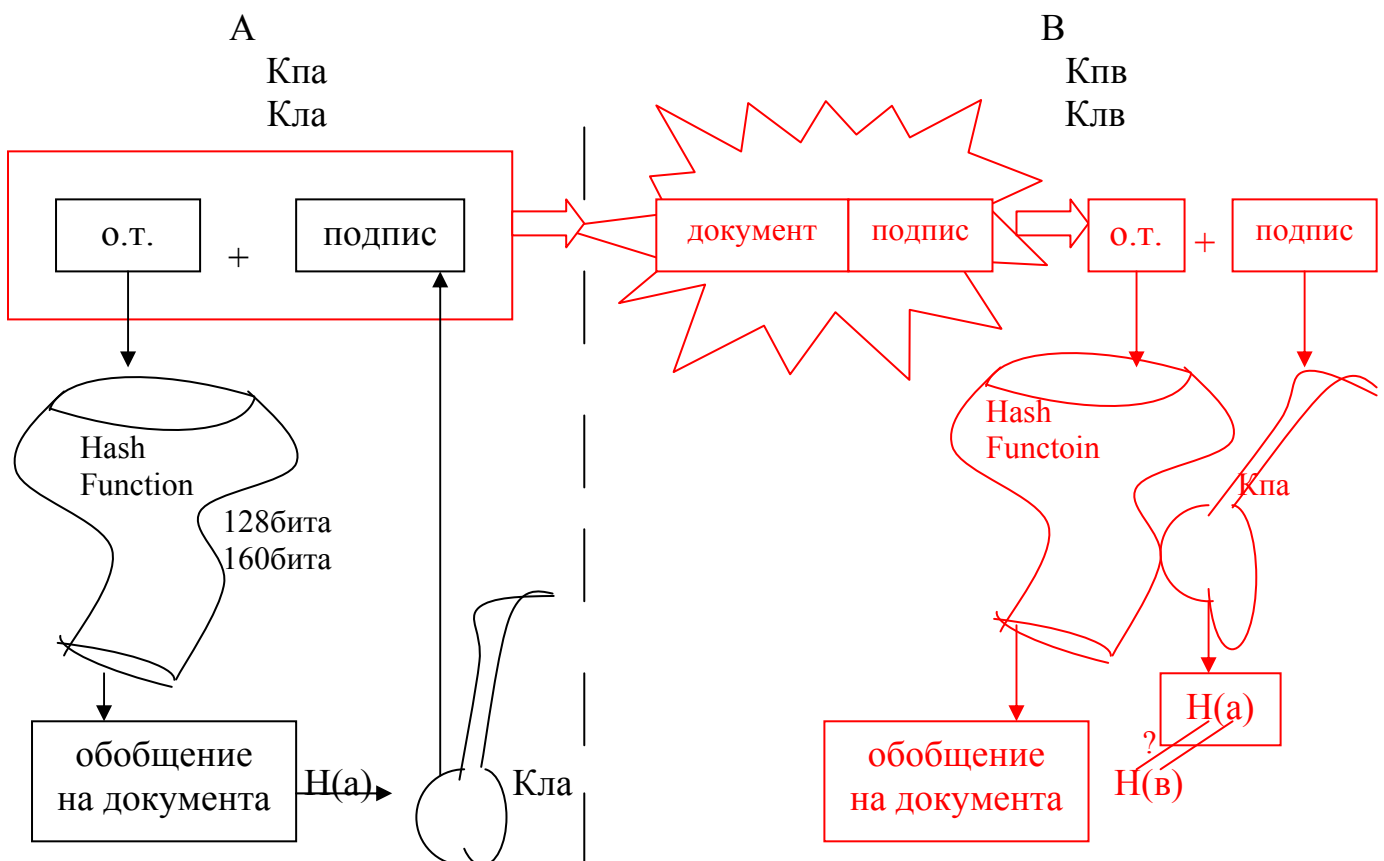


Р - Кпр, Клр

Но така не се постига запазване на съобщенията в тайна от другите.

Размера на ключа е 512, 1024, 2048, 4096, 8192 бита.

Процеса е доста по-бавен и тежък.



$H(a)$  и  $H(b)$  могат да са равни, еднакви, това означава, че пристигналият открит текст е еднакъв с автентичния открит текст, това означава, че документа не е променян при пътуването.

Hash функции – математическо изчисление приложено в съобщението за да се генерира малък низ наречен обобщение на документа, което представлява целия документ или файл. С всеки хеш код се създава уникално обобщение на документа. Два еднави документа притежават еднакви обобщения, но ако дори 1 бит е променен се различава съществено. Уникалността на съобщението зависи от създателя на документа на базата на заложените примитивни механизми в хеш функции.

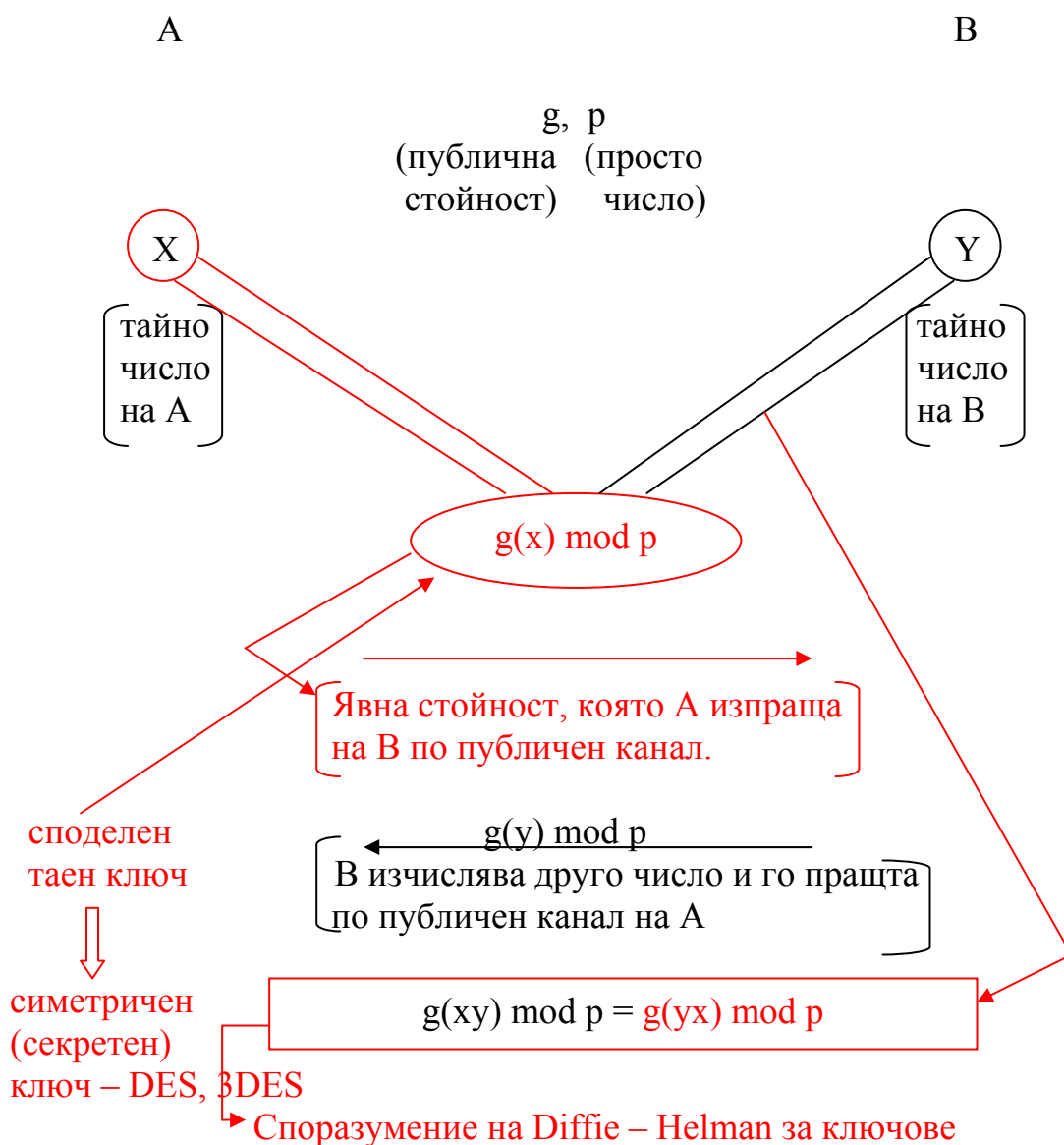
Невъзможно е да се създадът два еднакви подписа, обобщения на различни документи.

Всяко обобщение на документ криптирано със съобщение на съответния частен ключ е уникално за създателя си и може да се разчете само със съответния публичен ключ. Потвърждава се и целостта на обменяните документи.

14. 04. 2008 г.

Diffie – Helman споразумение за ключове

Diffie – Helman – основоположници на симетричната криптография – 1975 г., Stanford University. Те прилагат механизъм за криптиране, който използва два различни ключа. Същинската цел на този механизъм е разработване на метод за обмен на симетрични, секретни ключове при комуникация между две страни при несигурна мрежа. Но все пак разработения от тях механизъм не може да се нарече метод в истинския смисъл на думата. Разработването им се базира на математически функции, което позволява да се генерира т.н. споделен таен ключ. Той се ползва за криптиране на секретния таен ключ, който може да се използва при DES, 3DES, и AES.



Споделения таен ключ се използва за криптиране и обмяне по мрежата на симетричния.

Функцията mod p осигурява генериране на двете страни с един и същи споделен ключ. Математическите функции и числото P са много трудни. Потенциалният нарушител не може да наруши това условие.

Споразумението за ключове се използва във всички организации – SSL, SSH, IPSec, TLS.

### RSA ( Rivest, Shamir, Adelman) алгоритъм за криптиране

Използват се числа и функции за генериране на тайни ключове.

Първата стъпка, която се извършва от двете страни е да се изберът две големи числа.

$a, b > 10$  на степен 100

Втората стъпка е да се изчисли стойността на  $n=a.b$ .

Третата стъпка е да се изчисли стойността  $x = (a-1).(b-1)$ .

Четвъртата стъпка е да се избере число  $d$ , което е относително просто за  $x$ .

И петата стъпка е да се изчисли  $e.d=1 \text{ mod } x$ .

Явния текст се разделя на блокове –  $p$  на брой блока в интервала от единици до  $n$ .

P:  $X(1), X(2), \dots, X(p)$

$1 < p < n$

В процеса на криптиране  $E=P.e \text{ (mod } n)$ .

В процеса на декриптиране  $P=E.d \text{ (mod } n)$ .

Независимо, че имаме публичен и частен ключ на базата на тези функции при преобразуването се запазва криптираната информация.

### RSA обмен на ключове = Diffie – Hellman обмен на ключове



Обмен на симетричния (секретен) ключ на DES, 3DES, AES и въобще на алгоритми работещи със секретен ключ.

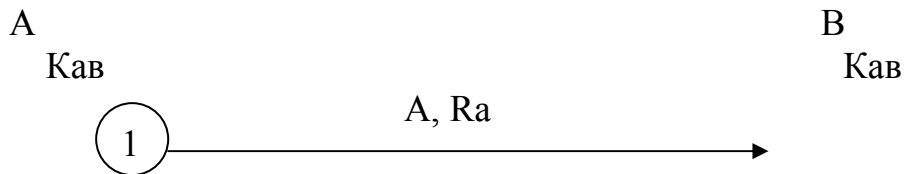
Секретния ключ се криптира с публичния ключ на получателя и се декриптира с личния ключ на получателя.



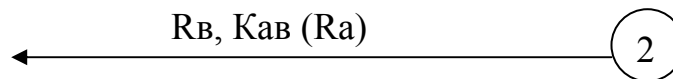
В криптира това число и го праща на А, така вече са сигурни, че това са двете страни, които трябва да си обменят съобщения.

Този протокол се нарича петстъпков и се отнася за автентификацията.

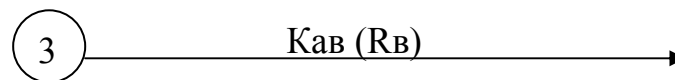
Има и тристъпков модел.



А изпраща съобщение с името си до В.



В връща своето случайно число и криптира случайното число на А, така криптирайки числото А знае, че срещу нея е В.



Когато В получи декриптирано своето число вече е сигурен, че това е А.

Но този тристъпков модел не е много надежден, защото В при това разкрива самоличността си преди да е сигурен, че това е А.

### **КДС (Key Distribution Center) – механизъм за автентификация**

При този подход всеки потребител съхранява свой секретен ключ в КДС. От своя страна КДС е отговорен за автентификация и управление на сесийните ключове. Най-простия протокол е известен под името wide-mouth frog – широка жабешка уста.



А обявява, че иска да кореспондира с В пред КДС, както и своята самоличност, като всичко това е криптирано с тайния ключ на А. КДС декриптира това съобщение, за да вземе самоличността на В и сесийния ключ. След това повторно криптира и изпраща на В самоличността на А и предложението от А сесийен ключ. В този случай проверката за самоличността е поверена на КДС.

## Kerberos

Това е протокол за автентификация, който позволява на потребителите взаимодействащи по мрежа да удостоверяват самоличността си. Този протокол извършва проверката за автентификация чрез така наречените независими услуги. Системи на които е инсталиран Kerberos изискват потребителя да напише само веднъж паролата си и всички проверки за автентификация се правят от Kerberos.

Първия продукт се е провел в MIT 1987 г., като се е развил до стандартизиран продукт и използва най-честите системи. Версия 5 е вградена и се ползва за автентификация в Ms Windows 2000 и следващите версии. В приложения – FTP, Telnet rlogin. Всички програми и машини, които работят с Kerberos се наричат укрепени или керберизирани машини.

Компоненти на Kerberos:

- услуга за автентификация – проверява автентичността на потребителите по време на регистрацията;
- услуга за издаване на билети – издава билети, които са доказателство за автентичността самоличността на потребителите;
- сървари за приложения – предоставят услуги за клиента.

Тези три ключови услуги работят съвместно, за да предложат услуги за автентификация. Kerberos ползва и КДС дистрибуторния информационен център – обединява услугите за самоличност и издаване на билети.

КДС споделя таен ключ за всеки потребител и услуги в мрежата. За целта поддържа база данни за всички потребители за това и КДС е наречен база данни за Kerberos.

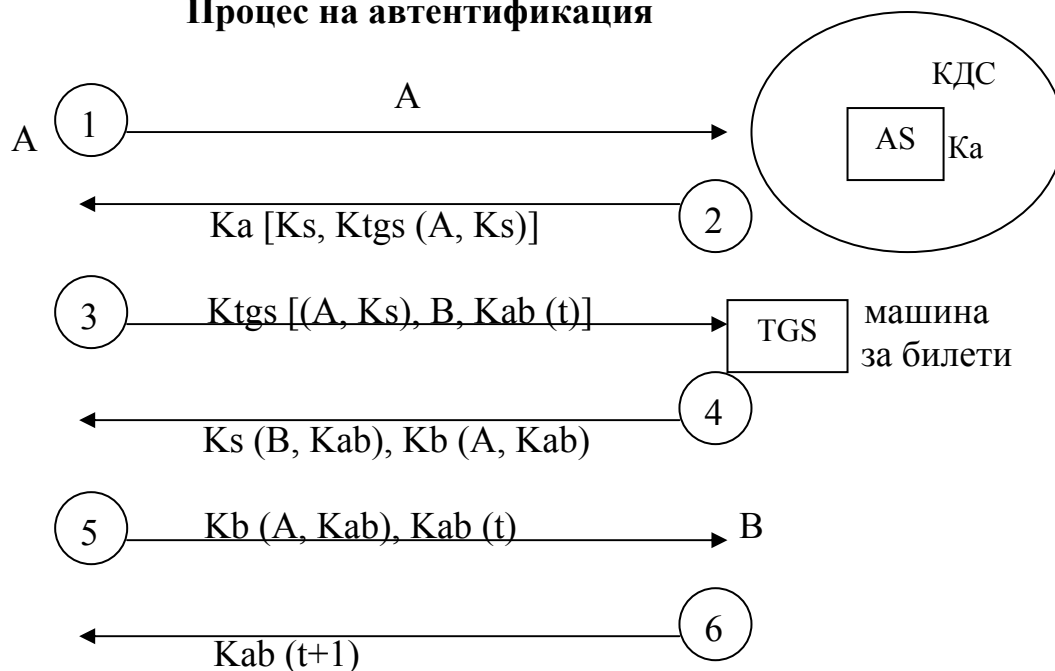
Сесионния ключ се генерира от КДС по случаен начин и се споделя между КДС и потребителя. Сесионния ключ съществува докато съществува и придружаващия го билет.

Kerberos използва т.н. билети, за да провери идентичността на основните единици в мрежата. Билетите проверяват сесионния ключ, самоличността на потребителя, идентификатор на услуги и IP адреса на клиента.

Сесионен ключ + билет + удостоверение за самоличност.

Индивидуална керберизирана услуга – автентифициран потребител получава билет за тази услуга.

## Процес на автентификация



В машината на А е инсталиран софтуер на Kerberos. А изпраща услуга към машината за автентификация. Машината отговаря като праща удостоверение за самоличност. А може да декриптира това съобщение, за да получи сесияния ключ и билет. След като успешно се идентифицира А в системата изпраща заявка до системата за билет да се издаде и на В. Посочва се и времето за комуникация между А и В. А получава сесияен ключ и възможност да комуникира с В. Накрая А комуникира с В и му праща информацията която има до този момент. И за да се затвори цикъла В уведомява, че е получил информацията.

## Цифров подпис

Има два типа RSA цифров подпис и DSA (Digital Signature Algorithm) цифров подпис. Механизма за цифров подпис е предложен от Сащ и се предлага за стандарт от американското правителство.

RSA – генериране на обобщение на документ на подпис и криптиране на това обобщение, и криптиране на частния ключ на подателя.

DSA – използват се специални математически функции, като се използват две 160 битови числа произхождащи от обобщението на документа и частния ключ на подателя. По аналогичен начин DSA използва публичния ключ на подателя за удостоверяване на подписа. Но това удостоверяване е много по сложен механизъм от RSA. DSA е много сложен и силен алгоритъм, има високо ниво на устойчивост от една страна, а от друга е бавен и натоварва процесора.