СПбГЭТУ «ЛЭТИ»
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ

А.Б. Левина

Криптография и криптографические протоколы

Вводная лекция

СПбГЭТУ «ЛЭТИ», 2022 г.

ВВЕДЕНИЕ

1.1 СОДЕРЖАНИЕ КУРСА

Данный курс будет посвящен основам криптографии, мы изучим следующие понятия и алгоритмы:

1. Исторические шифры.
2. Современные алгоритмы шифрования:
    - теоретико-информационная стойкость;
    - симметричные алгоритмы шифрования (алгоритм DES, AES, поточные шифры (GSM стандарт));
    - асимметричные алгоритмы шифрования и математические задачи, на которых они основываются (алгоритм RSA, криптосистема рюкзак);
    - гибридные криптосистемы.
3. Алгоритмы подписи.
4. Криптографические хэш-функции.
5. Криптоанализ.

1.2 СОДЕРЖАНИЕ ЛЕКЦИИ

На этой лекции мы начнем наше знакомство с криптографией и первое, что мы изучим - это исторические шифры. Данная лекция будет состоять из следующих фрагментов:

- введение, основные понятия;
- докомпьютерные шифры - **шифр Цезаря, шифр замены**;
- ненадежность докомпьютерных шифров и примеры атак на них;
- принцип работы роторной машины «Энигма»;

- понятия перестановки и замены как основных компонентов шифра.

2 ОСНОВНЫЕ ПОНЯТИЯ

Несмотря на повсеместное использование криптографии и реализации ее во всех современных устройствах, сама наука криптография является одной из самых древних наук, и появилась одномоментно с письменностью. История криптографии насчитывает более четырех тысяч лет.

Сам термин **"криптография"** состоит из слов "kryptos"– тайный, "grapho"– пишу и обозначает **"тайнопись".**

Рассмотрим основные определения, которые потребуются нам в дальнейшем.

Определение 2.0.1 *Криптография - наука о методах обеспечения конфиденциальности, целостности данных, аутентификации.*

Давайте рассмотрим каждое из этих понятий отдельно:

Определение 2.0.2 *Конфиденциальность – невозможность прочтения информации посторонним.*

Определение 2.0.3 *Целостность данных - невозможность незаметного изменения информации.*

Определение 2.0.4 *Аутентификациия – процедура проверки подлинности.*

Введем еще одно определение, которое нам понадобится.

Определение 2.0.5 *Шифрование – обратимое преобразование информации с помощью криптографического алгоритма и секретного ключа в целях сокрытия данной информации от неавторизованных лиц, с предоставлением, в это же время, авторизованным пользователям доступа к ней.*

Как видно из определений, конфиденциальность может быть обеспечена посредством шифрования.

Когда мы будем рассматривать криптографические алгоритмы, у нас обычно будет два пользователя, обменивающиеся информацией - Alice/Алиса (A) и Bob/Боб (B) и

атакующий – объект, который пытается получить информацию, не имея на то прав, Eva/Ева (E).

Для создания алгоритмов, которые Eva не сможет взломать, нам необходимо познакомиться с криптоанализом и криптостойкостью.

Определение 2.0.6 *Криптоанализ (от др.-греч. "kryptos"«скрытый» + «анализ») — наука о методах дешифровки зашифрованной информации без предназначенного для этого ключа, а также сам процесс такой дешифровки.*

Определение 2.0.7 *Криптостойкость - способность криптографического алгоритма противостоять криптоанализу.*

Часто, при изучении криптографии, у студентов возникает путаница между криптографией и стеганографией, давайте введем понятие стеганографии. Несмотря на то, что данный раздел науки очень интересен и сам по себе, мы, в нашем курсе, его касаться не будем.

Определение 2.0.8 *Стеганография — это наука о скрытой передаче информации путём сохранения в тайне самого факта передачи.*

Термин был введен в 1499 году Иоганн Тритемий в своем трактате «Стеганография».

После введения основных определений, мы можем уже ввести обозначения, которые будут использоваться нами на протяжение всего курса.

3 ОБОЗНАЧЕНИЯ И ЭТАПЫ РАЗВИТИЯ КРИПТОГРАФИИ

3.1 ОБОЗНАЧЕНИЯ

К сожалению, в криптографии нет устоявшихся обозначений, поэтому мы будем использовать обозначения, используемые в англоязычной литературе.

Рассмотрим процесс шифрования. Мы помним, что шифрование – это обратимое преобразование информации с помощью криптографического алгоритма и секретного ключа в целях сокрытия данной информации от неавторизованных лиц. Нам нужно преобразовать сообщение так, чтобы атакующий не смог его прочесть, введем следующие обозначения:

шифрующая функция;

D – $Decrypt$ - расшифровывающая функция;

C – $Cipher$ - шифротекст, зашифрованный текст;

m – $message$ - открытый текст, то есть текст, который нам необходимо зашифровать;

k – key - секретный ключ, который мы будем использовать для шифрования и дешифрования.

В описанных обозначениях, процесс шифрования может быть представлен следующим образом:

$$C = E_k(m),$$

то есть шифротекст C получен в результате шифрования открытого текста m посредством шифрующей функции E с использованием ключа k.

Если один объект шифрует открытый текст/сообщение, то у нас должен быть второй объект, который может эту информацию дешифровать и получить то сообщение, которое шифровалось. Это делается с помощью дешифрующей функции D, тогда

$$m = D_k(C).$$

Введем два определения, которые будут с нами на протяжении всего курса.

Определение 3.1.1 *Криптосистемы называются симметричными криптосистемами, или криптосистемами с секретным ключом, если процессы шифрования и дешифрования используют один и тот же ключ.*

Определение 3.1.2 *Криптосистемы называются асимметричными криптосистемами, или криптосистемами с открытым ключом, если процессы шифрования и дешифрования используют два различных ключа.*

С работой симметричных и асимметричных криптосистем мы более детально познакомимся позже, а сейчас давайте начнем наше знакомство с этапами развития криптографии и рассмотрим самые известные исторические шифры.

3.2 ЭТАПЫ РАЗВИТИЯ КРИПТОГРАФИИ

Так как криптография является довольно древней наукой, ее развитие можно

разделить на несколько этапов:

Как мы видим, все 4000 лет можно разделить на три основных этапа:

- «Ручной период» – когда шифрование производилось вручную, было очень нестойким и вся криптостойкость алгоритма обеспечивалась за счет содержания в секрете алгоритма шифрования;
- момент появления электромеханических устройств. Шифрование посредством таких устройств обеспечивало некоторый уровень крипто- стойкости;
- современный период. К этому периоду мы можем отнести момент появление компьютерной криптографии.

Если мы будем рассматривать «Ручной период», то его можно разделить на два этапа:

- моноалфавитные шифры - методы шифрования, которые сводятся к созданию специальных таблиц шифрования, в которых для каждой буквы открытого текста существует единственная соответствующая ей буква шифротекста.
- полиалфавитные шифры - методы шифрования, в которых буквы открытого текста заменяются различными буквами шифротекста, в зависимости от позиции буквы в открытом тексте.

Как раз с моноалфавитными, полиалфавитными шифрами и электроме- ханическими устройствами мы сейчас и познакомимся.

4 ИСТОРИЧЕСКИЕ ШИФРЫ

4.1 ШИФР ЦЕЗАРЯ

Самый первый алгоритм шифрования, с которым мы познакомимся – это **шифр Цезаря**. Шифр Цезаря является моноалфавитным шифром, в котором каждый символ открытого текста заменяется символом, находящимся тремя символами правее. Пример представлен на рисунке.

Рис.: Пример таблицы соответствия символов в шифре Цезаря.

Например, при шифровании с помощью алгоритма Цезаря, слово **hello** перейдет в **KHOOR.**

Пример шифрования при помощи шифра Цезаря.

Шифр Цезаря очень легко реализуем, но он является абсолютно не криптостойким, о чем мы поговорим чуть позже.

4.2 ШИФР ЗАМЕНЫ

На смену шифру Цезаря пришел **шифр замены**. Если в шифре Цезаря шифрование обеспечивалась путем простого сдвига, то в шифре замены шифрующий алфавит представляет из себя случайную последовательность букв, например как представлено на рисунке:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
L	F	W	P	B	Y	E	Q	J	X	C	T	K	O	G	V	A	S	R	H	U	N	D	Z	I	M

Пример шифра замены.

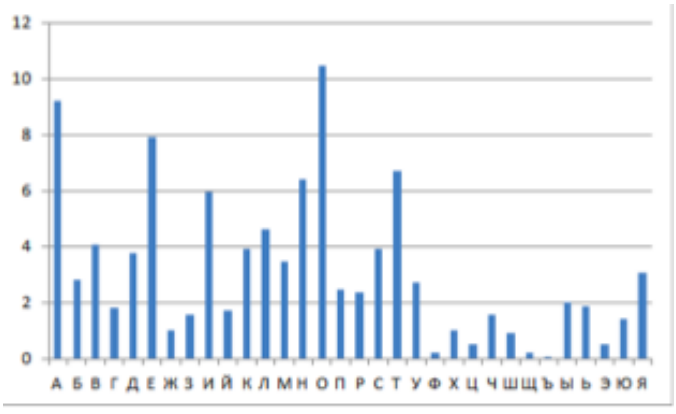
При шифровании с помощью представленного шифра замены слово **hello** перейдет в **QBTTG**.

Как мы видим, и шифр Цезаря, и шифр замены являются моноалфавитными шифрами, так как каждая буква открытого текста всегда меняется на одну и ту же букву шифрующего алгоритма. Такие шифры поддаются «взлому» с помощью частотного анализа, посмотрим, что это такое.

4.3 ЧАСТОТНЫЙ АНАЛИЗ

В каждом алфавите буквы в тексте встречаются с разной частотой, например, для русского языка самой часто встречающейся буквой является буква **о**. Зная частоту появления букв в тексте и видя шифротекст, полученный с помощью моноалфавитного алгоритма шифрования, можно сразу догадаться, какая буква открытого текста соответствует какой букве шифротекста.

На рисунке представлено распределение частот появления букв русского алфавита в тексте из 100 символов.



Частотный анализ букв русского алфавита.

Когда происходит взлом сообщения, зашифрованного моноалфавитным шифром, мы смотрим какие самые часто встречаемые буквы есть в шифротексте и начинаем сравнивать их статистику со статистикой языка, таким образом понимая, в какую букву шифротекста какая буква открытого текста перешла. При этом очень поможет при взломе также знать статистку появления биграмм и триграмм (самые часто встречающиеся сочетания из двух и трех букв).

Как видно, моноалфавитные шифры абсолютно не надежны и нам надо передвигаться вперед, к более криптостойким алгоритмам.

4.4 ПОЛИАЛФАВИТНЫЕ ШИФРЫ, ШИФР ВИЖЕНЕРА (ВИЖНЕРА)

Шифр Виженера является одним из самых известных исторических полиалфавитных шифров и пришел на замену моноалфавитным шифрам. Данный шифр способен противостоять частотному анализу. Рассмотрим работу полиалфавитных шифров.

В отличие от моноалфавитных шифров, в полиалфавитных шифрах используется несколько шифрующих алфавитов, поэтому получается, что отсутствует закрепление за буквой открытого текста единственной буквы шифрующего алгоритма. Таким образом, буква открытого текста будет менять свое значение уже в зависимости от своей позиции в открытом тексте. Мы можем увидеть это на примере, использующем два шифрующих алфавита, где буквы, стоящие на четных позициях, шифруются с помощью одного шифрующего алфавита, а буквы, стоящие на нечетных позициях, – с помощью другого.

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Нечетная позиция	L	F	W	P	B	Y	E	Q	J	X	C	T	K	O	G	V	A	S	R	H	U	N	D	Z	I	M
Четная позиция	J	P	F	Z	N	C	B	T	G	O	Y	K	R	E	V	M	H	L	S	X	I	U	W	A	Q	D

hello→ QNTKG (а не QBTTG)

Пример использования полиалфавитного шифра.

Прежде чем переходить к одному из самых известных полиалфавитных шифров – шифру Виженера, давайте вспомним одну математическую операцию, операцию взятие по модулю (mod), или, как ее еще называют, деление с остатком. Деление с остатком – это арифметическая операция, часто используемая в криптографии, данная операция

определена для целых или натуральных чисел следующим образом.

Определение 4.4.1 *Пусть* a *и* b *- целые числа, причём* $b = 0$. *Деление с остатком* a *на* b *означает нахождение таких целых* q *и* r, *для которых выполняется равенство:*

$$a = b \cdot q + r, \quad r \in \{0, 1, \ldots, b - 1\}.$$

В описанном представлении число r *называется остатком от деления* a *на* b.

Замечание 4.4.1 *В обозначениях предыдущего определения, то, что число* r *является остатком от деления* a *на* b, *часто записывают так:*

$$r = a \mod b.$$

Приведем пример: $37 = 12 \cdot 3 + 1$, это записывается как $37 \mod 12 = 1$.

Рассмотрим теперь шифр Виженера.

Шифр Виженера использует идею полиалфавитного шифрования следующим образом: выбирается какое-то секретное слово, которое как раз является ключом и это слово «складывается» с открытым текстом. Сформулируем алгоритм более четко.

Пусть - алфавит, состоящий из n элементов (символов), номер символа алфавита будем обозначать i, z_i - символ, стоящий на i-ой позиции. Например, при стандартной нумерации букв в русском алфавите, z_0 = а, z_1 = б, z_{32} = я.

Обозначим через m открытый текст, то есть текст, подлежащий шифрованию. Пусть m - длина открытого текста, m_j - буква открытого текста, стоящая на j ой позиции в открытом тексте, а m_j - номер, отвечающий букве m_j в алфавите;

Рассмотрим пример. Пусть - стандартно пронумерованный русский алфавит, открытый текст m - это слово криптография, m = 12. Во введенных обозначениях, $m_8 = a$, $j = 8$, а $|m_8| = 0$.

Пример 4.4.1 *Зашифруем слово* m = *криптография с помощью ключа* k = *шифр в рамках стандартно пронумерованного русского алфавита. Так как в*

нашем случае длина ключа $k = 4$*, алфавит состоит из* $n = 33$ *элементов, то первая буква* c_0 *шифротекста ищется следующим образом:*

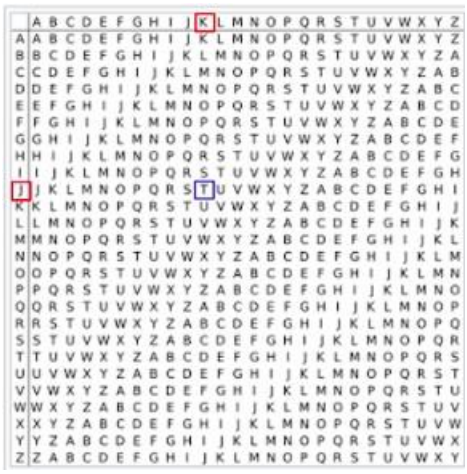
$$c_0 = z_l, \qquad l = (|m_0| + |k_{0 \mod 4}|) \mod 33.$$

m_0 *- это буква «к», в алфавите она имеет номер* 11*. Второе слагаемое в скобках - номер в алфавите соответствующего символа ключа. Так как ключ может быть короче, чем шифруемый текст, мы его циклически «копируем» до достижения нужной длины, отсюда и индекс, включающий понятие остатка.*

Для реализации своего алгоритма Виженер предложил использовать «квадрат Виженера».

В данном квадрате в верхней строке и в левом столбце записаны алфавиты; для шифрования открытого текста мы находим в верхней строке букву открытого текста, которую мы шифруем, а в левом столбце находим букву ключа (нумерация начинается с 0), с помощью которого происходит шифрование, и смотрим, какая буква в квадрате стоит на пересечении данной строки и данного столбца. Буква, которая находится на пересечении, как раз и будет буквой нашего шифротекста. Посылающий сообщение записывает ключевое слово циклически до тех пор, пока его длина не будет соответствовать длине исходного текста.

Если мы рассмотрим на примере, то увидим, что буква **K** открытого текста «складываясь» с буквой ключа **J** на выходе даст нам букву **T**.



Так мы проходимся по каждой букве открытого текста и получаем шифротекст. Но, несмотря на кажущуюся криптостойкость, шифр Виженера тоже был взломан, алгоритм взлома шифра Виженера назван в честь Фридриха Вильгельма Касиски, офицера прусской

армии, который разработал данный алгоритм в 1863 году, хотя некоторые работы по взлому шифра Виженера были предложены и ранее.

4.5 ТЕСТ КАСИСКИ

Тест Касиски основан на нахождении длины секретного слова – ключа, так как если мы сможем найти длину ключа, то, соответственно, будем знать сколько шифрующих алфавитов использовалось, а тогда, зная количество этих алфавитов, будет известно на каких позициях используются те же алфавиты и, соответственно, на этих позициях статистика языка будет оставаться.

Как только длина ключевого слова обнаружена, криптоаналитик выстраивает зашифрованный текст в n колонках, где n — длина ключевого слова. Тогда каждую колонку можно рассматривать как зашифрованный моноалфавитным шифром текст, который можно подвергнуть частотному анализу.

Метод Касиски заключается в поиске групп символов, которые повторяются в зашифрованном тексте. Группы должны состоять из не менее чем трех символов. Тогда расстояния между последовательными возникновениями групп, вероятно, будут кратны длине ключевого слова. Предполагаемая длина ключевого слова кратна наибольшему общему делителю всех расстояний.

После нахождения наибольшего общего делителя мы делим текст на блоки данной длины и, используя частотный анализ, находим открытый текст.

На смену шифру Вижнера пришел новый алгоритм шифрования - перестановочный шифр, который мы сейчас рассмотрим.

4.6 ПЕРЕСТАНОВОЧНЫЙ ШИФР

Определение 4.6.1 *Перестановочный шифр, или шифр перестановки — это метод симметричного шифрования, в котором символы открытого текста меняются местами согласно заданной заранее перестановке.*

Элементами текста могут быть отдельные символы (самый распространенный случай), пары букв, тройки букв, комбинирование этих случаев и так далее. Рассмотрим более подробно работу перестановочных шифров, т.к. идеи, лежащие в основе шифра перестановки, используют и современные крипто- алгоритмы. Позже мы увидим, что в шифрах DES и Rijndael присутствуют компоненты, называемые S-блоками, которые являются простыми

подстановками, а другие составляющие современных симметричных шифров основываются на перестановках.

Перестановочные шифры активно применялись в течение нескольких столетий, опишем принцип их работы.

Фиксируется длина ключа n, текст разбивается на блоки длины n и в каждом блоке фиксируется перестановка S_n, именно перестановка является секретным ключом.

Например, зафиксируем следующую перестановку:

$s =$

1	2	3	4	5
2	4	1	3	5

Соответственно, здесь **1-ый** символ нашего открытого текста пойдет на **2-ую** позицию, **2-ой** символ на **4-ую** позицию, **3-ий** на **1-ую** позицию, **4-ый** на **3-ию** позицию, а **5-ый** останется на **5-ой** позиции.

Если мы рассмотрим это на примере, то увидим:

1. *once upon a time there was a little girl called Snow White*
2. *onceu ponat imeth erewa salit tlegi rlcal ledsn owwhi te*
3. *coenu npaot eitmh eewra lsiat etgli crall dlsen wohwi et*
4. *Coenunpaoteitmheewralsiatetglicralldlsenwohwiet*

Здесь шаг 1 — исходный текст, шаг 2 — исходный текст, разбитый на блоки длиной пять, шаг 3 -перестановки внутри блоков, шаг 4 -убираем пробелы и получаем шифротекст.

Но и данный шифр можно взломать. Для его взлома нам понадобится **атака с выбором открытого текста**. В данной атаке атакующий «подсовывает» законному пользователю свой текст и просит его зашифровать. Имея на руках открытый текст и зашифрованный, атакующий сразу понимает, какая перестановка использовалась.

Давайте рассмотрим пример.

Пример 4.6.1 *Алиса использует перестановку* $S_4 = 3, 1, 4, 2$. *Ева подсовывает ей текст crypto is good, посмотрим как Алиса будет проводить шифрование: 1. Разбиение на блоки: cryp tois good*

2. Проведение перестановок в каждом блоке: ycpr itso ogdo

3. Шифротекст будет - ycpritsoogdo

Посмотрим как Ева будет восстанавливать ключ из данного шифротекста:

1. У Евы есть открытый текст crypto is good и шифротекст ycpritsoogdo.

2. Ева проводит анализ и видит, что буква c, которая была на первой позиции, в зашифрованном тексте находится на второй, то есть в ключе произошла перестановка (1 → 2). Буква r со второй позиции ушла на четвертую, то есть произошло (2 → 4), y на 1 (3 → 1), p на 3 (4 → 3),

t на 6 (5 → 6), если она пройдется так по всем буквам она восстановит последовательность $S_4 = \{3, 1, 4, 2\}$.

На этом мы заканчиваем знакомство с «Ручным периодом» шифрования и переходим к работе роторных машин.

Соответственно, здесь **1-ый** символ нашего открытого текста пойдет на **2-ую** позицию, **2-ой** символ на **4-ую** позицию, **3-ий** на **1-ую** позицию, **4-ый** на **3-ию** позицию, а **5-ый** останется на **5-ой** позиции.

Если мы рассмотрим это на примере, то увидим:

1. once upon a time there was a little girl called Snow White
2. onceu ponat imeth erewa salit tlegi rlcal ledsn owwhi te
3. coenu npaot eitmh eewra lsiat etgli crall dlsen wohwi et
4. Coenunpaoteitmheewralsiatetglicralldlsenwohwiet

Здесь шаг 1 — исходный текст, шаг 2 — исходный текст, разбитый на блоки длиной пять, шаг 3 -перестановки внутри блоков, шаг 4 -убираем пробелы и получаем шифротекст.

Но и данный шифр можно взломать. Для его взлома нам понадобится **атака с выбором открытого текста**. В данной атаке атакующий «подсовывает» законному пользователю свой текст и просит его зашифровать. Имея на руках открытый текст и зашифрованный, атакующий сразу понимает, какая перестановка использовалась.

Давайте рассмотрим пример.

1. Разбиение на блоки: cryp tois good

4. Проведение перестановок в каждом блоке: ycpr itso ogdo

5. Шифротекст будет - ycpritsoogdo

Посмотрим как Ева будет восстанавливать ключ из данного шифротекста:

3. У Евы есть открытый текст crypto is good и шифротекст ycpritsoogdo.

4. Ева проводит анализ и видит, что буква c, которая была на первой позиции, в зашифрованном тексте находится на второй, то есть в ключе произошла перестановка (1 → 2). Буква r со второй позиции ушла на четвертую, то есть произошло (2 → 4), y на 1 (3 → 1), p на 3 (4 → 3), t на 6 (5 → 6), если она пройдется так по всем буквам она восстановит последовательность S_4 = {3, 1, 4, 2}.

На этом мы заканчиваем знакомство с «Ручным периодом» шифрования и переходим к работе роторных машин.

Сверху «Энигма» выглядела как пишущая машинка, а внутри был сильнейший криптографический алгоритм. Рассмотрим ее внутреннее устройство:

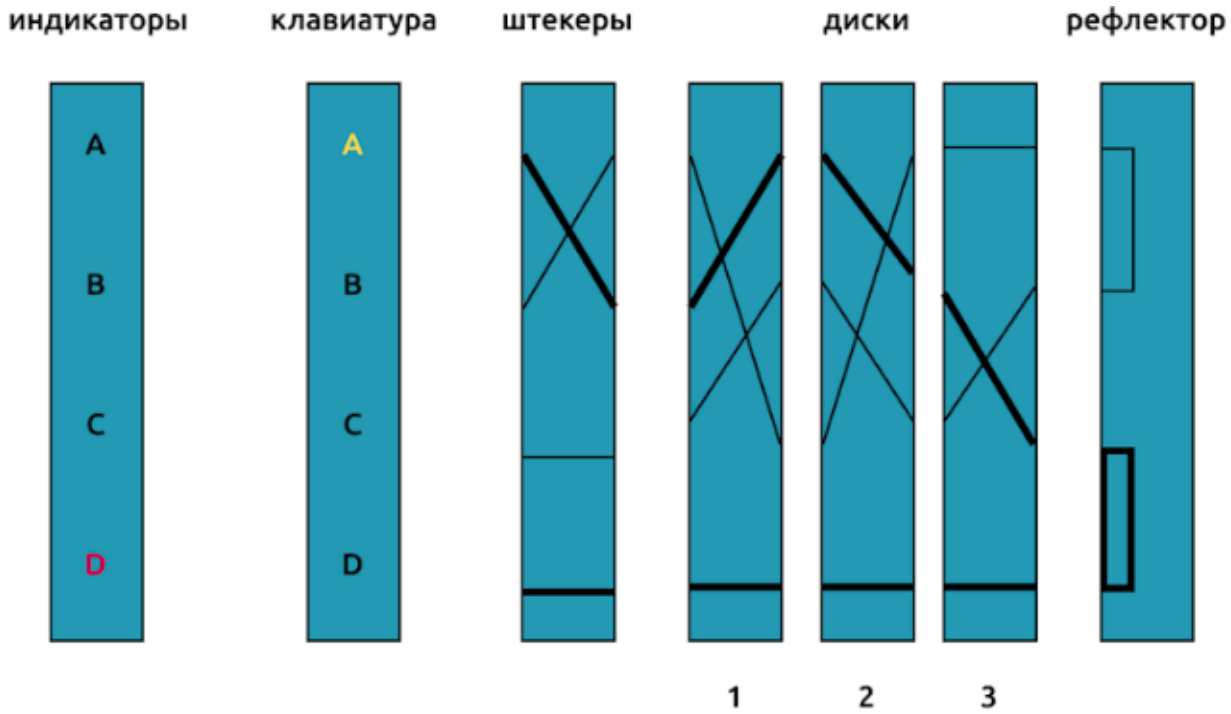
1. Сверху шли индикаторы и клавиатура, это как раз была пишущая машинка, которая обеспечивала набор текста.

2. Штекеры обеспечивали замену: буквы открытого текста менялись на буквы первого шифрующего алфавита по заданному порядку.

3. Три диска ротора обеспечивали основную работу шифрования.

4. Рефлекторы обеспечивали обратную связь и давали возможность дешифроваться.

Схема работы Энигмы представлена на рисунке.



Ключом в «Энигме» было:

1. Расположение штекеров.

2. Коммутационные диски и их компоновка.

3. Позиции колец.

4. Начальное угловое положение дисков.

Перед передачей приходил так называемый **сеансовый ключ**, который давал возможность настроить принимающей стороне все компоненты машины для сеанса связи. Изначально предполагалось, что ключ будет меняться каждый сеанс связи, но обеспечить это не было возможным, поэтому ключ менялся только раз в день, тем самым у криптоаналитиков была возможность в течение дня взломать алгоритм и установить используемый ключ.

5.1 КОМПОНЕНТЫ ИСТОРИЧЕСКИХ ШИФРОВ

При знакомстве с историческими шифрами мы познакомились с такими понятиями как **замена** и **перестановка**.

- Замены (сдвиг – частный случай замены) (**substitution**);
- Перестановки (**permutation**).

5 ЗАКЛЮЧЕНИЕ

В этой лекции мы познакомились с основными понятиями, которые используются в криптографии, ввели обозначения и определения, необходимые нам на протяжении всего курса, рассмотрели исторические шифры, атаки на них, изучили устройство и работу роторных машин и ввели понятие перестановки и замены, которые используются при построении современных шифров.