

ФГБОУ ВО «Воронежский государственный технический университет»

Кафедра систем автоматизированного проектирования
и информационных систем

**РАЗРАБОТКА ПРОГРАММНЫХ СРЕДСТВ
КРИПТОГРАФИЧЕСКОЙ ЗАЩИТЫ WEB-
КОНТЕНТА**

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
к лабораторным работам по дисциплине "Технологии
защиты web-контента" для студентов направления 09.03.02
«Информационные системы и технологии» очной формы
обучения



Воронеж 2019

Введение

Целью курса лабораторных занятий по дисциплине «Технологии защиты web-контента» является приобретение практических навыков решения задач защиты информации, возникающих при ее обработке, хранении и передаче по открытым каналам связи. В данном методическом пособии приведены задания к трем лабораторным работам, посвященным криптографическим методам защиты web-контента. В пособии подробно описаны алгоритм симметричного шифрования ГОСТ, основные алгоритмы теории чисел, используемые в криптографии, алгоритм шифрования с открытым ключом RSA.

В ходе выполнения каждого задания лабораторного практикума разрабатывается и отлаживается программный комплекс, написанный на одном из языков программирования. Программа должна иметь интуитивно понятный интерфейс, позволять изменять исходную и полученную в результате шифрования информацию. Результат работы оценивается в процессе тестирования программы на наборе контрольных примеров. Отчет по лабораторной работе должен содержать:

- титульный лист;
- задание;
- краткую теоретическую часть и алгоритм;
- фрагмент текста программы с реализованным алгоритмом;
- пример выполнения программы;
- вывод по проделанной работе.

Основные понятия

Криптографические алгоритмы шифрования используются для обеспечения конфиденциальности хранимых или передаваемых данных. Алгоритмы с помощью определенных правил преобразуют исходные данные в зашифрованный вид так, чтобы восстановить эти данные мог только законный пользователь (**шифрование**). Для получения исходной информации необходимо над зашифрованным текстом выполнить обратный процесс преобразования – **дешифрование**. При шифровании и дешифровании данных применяется сменный элемент алгоритма, называемый в криптографии **ключом**.

Рассмотрим классическую задачу передачи сообщений. Пусть участник А желает переслать секретное сообщение участнику В по открытому информационному каналу. В качестве участников такого взаимодействия могут выступать обычные пользователи, прикладные программы, объекты сетевой инфраструктуры, например, маршрутизаторы и др.

Исходное сообщение участника А называется **открытым текстом** и обозначается буквой М (от англ. message). Зашифрованное с помощью некоторого алгоритма сообщение называется **шифротекстом** и обозначается буквой С (от cipher text).

В **симметричных алгоритмах** для шифрования и дешифрования сообщений используется один общий для участников секретный ключ K_{AB} . Схема такого шифрования приведена на рисунке 1.

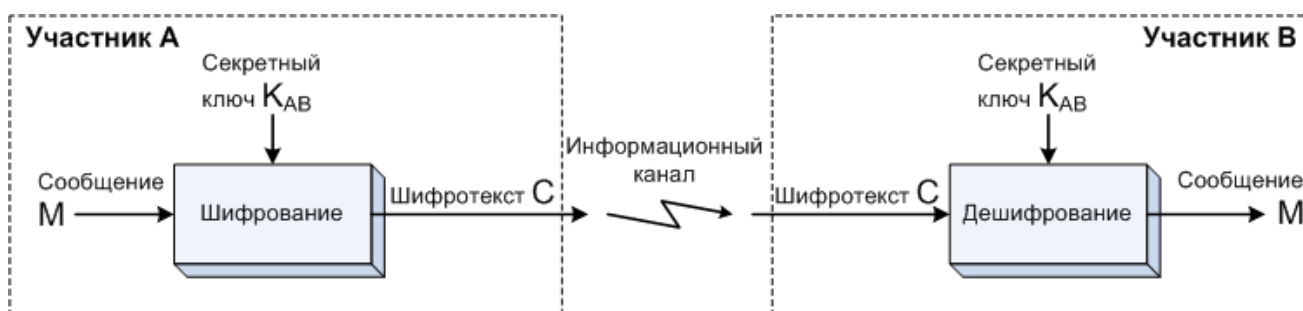


Рис. 1. Схема симметричного шифрования

В зависимости от способа обработки исходного сообщения различают потоковые и блочные алгоритмы симметричного шифрования. **Потоковые алгоритмы** обрабатывают исходное сообщение побитно (иногда небольшими группами бит, например, по 8 бит). **Блочные алгоритмы** работают с блоками открытого текста. Размер блока равен степени двойки, например, блок размером 64 бита.

Наиболее известными и используемыми алгоритмами симметричного шифрования являются DES, ГОСТ, Blowfish, IDEA, AES, Rijndael.

В **алгоритмах шифрования с открытым ключом** используется два ключа: открытый (публичный) ключ K^+ для шифрования и личный (секретный) ключ K^- для дешифрования сообщений. Схема шифрования с открытым ключом приведена на рисунке 2.

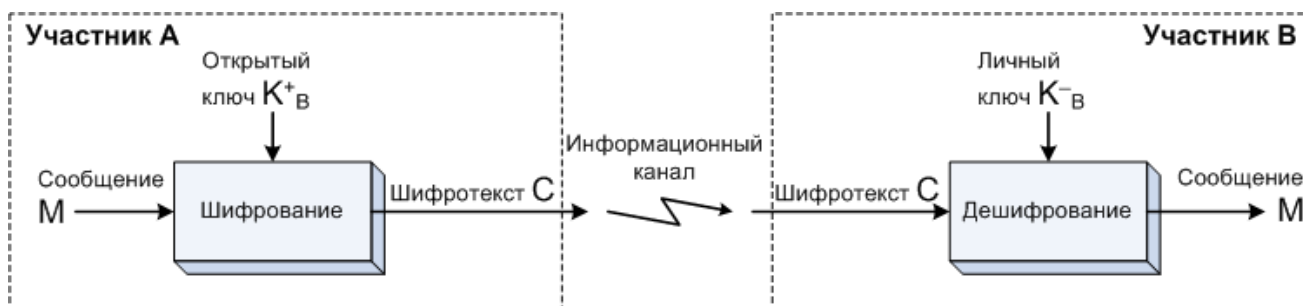


Рис. 2. Схема шифрования с открытым ключом

Наиболее практичными алгоритмами шифрования с открытым ключом являются RSA, Rabin, шифр Эль-Гамала.

Лабораторная работа № 1

«АЛГОРИТМЫ СИММЕТРИЧНОГО ШИФРОВАНИЯ»

Задание. Написать программу, реализующую алгоритм симметричного шифрования ГОСТ 28147-89. Режим выполнения алгоритма – простая замена.

Общие сведения. Алгоритм шифрования ГОСТ 28147-89 является симметричным, блочным алгоритмом. Преобразование осуществляется над блоком размером 64 бита, размер секретного ключа 256 бит, в алгоритме 32 раунда преобразований.

Необходимые определения и обозначения:

X – блок открытого текста размером 64 бита;

Y – блок зашифрованного текста размером 64 бита;

K – секретный ключ (256 бит);

W – раундовый ключ.

В алгоритме ГОСТ используются следующие операции:

S-блок или S-box – табличная подстановка, при которой группа бит отображается в другую группу бит;

\boxplus – операция сложения по модулю 2^{32} ;

\oplus или XOR – операция сложения по модулю 2 (или побитовое «исключающее или»);

$\leftrightarrow 11$ – циклический сдвиг влево на 11 бит.

Эти операции циклически повторяются в алгоритме, образуя так называемые раунды. Входом каждого раунда является выход предыдущего раунда и раундовый подключ W_i , который получен из секретного ключа шифрования K следующим образом. Рассмотрим секретный ключ K (256 бит), состоящий из восьми слов по 32 бита: $K = K_0K_1K_2K_3K_4K_5K_6K_7$. На их основе строим раундовый ключ W :

$$W = \underbrace{K_0K_1K_2K_3K_4K_5K_6K_7}_{\text{Раунды 1-8}} \underbrace{K_0K_1K_2K_3K_4K_5K_6K_7}_{\text{Раунды 9-16}} \underbrace{K_0K_1K_2K_3K_4K_5K_6K_7}_{\text{Раунды 17-24}} \underbrace{K_7K_6K_5K_4K_3K_2K_1K_0}_{\text{Раунды 25-32}}.$$

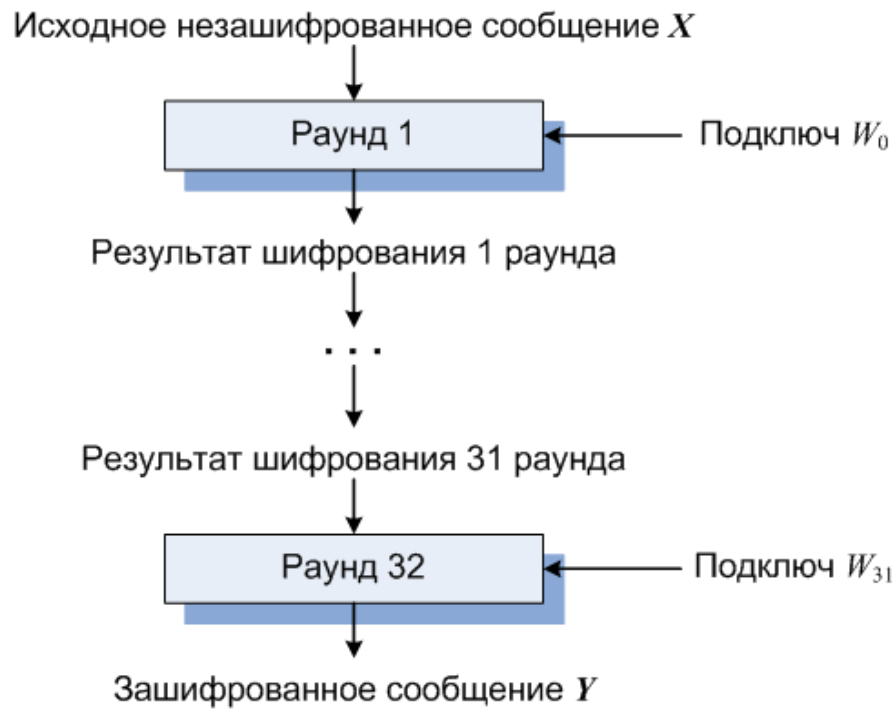


Рис. 3. Основная схема алгоритма ГОСТ

Для шифрования блок открытого текста сначала разбивается на две одинаковые части, правую R (младшее слово) и левую L (старшее слово).

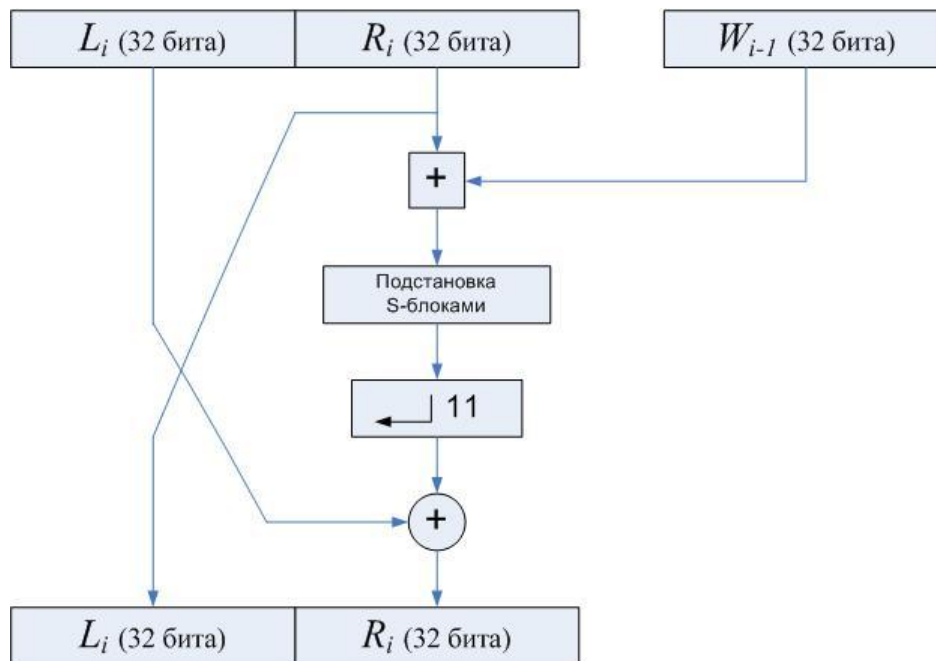


Рис. 4. Схема одного раунда алгоритма ГОСТ

На i -том раунде используется подключ W_{i-1} . Правая часть R_i складывается по модулю 2^{32} с раундовым подключом W_{i-1} . Над получившимся результатом выполняется операция табличной подстановки.

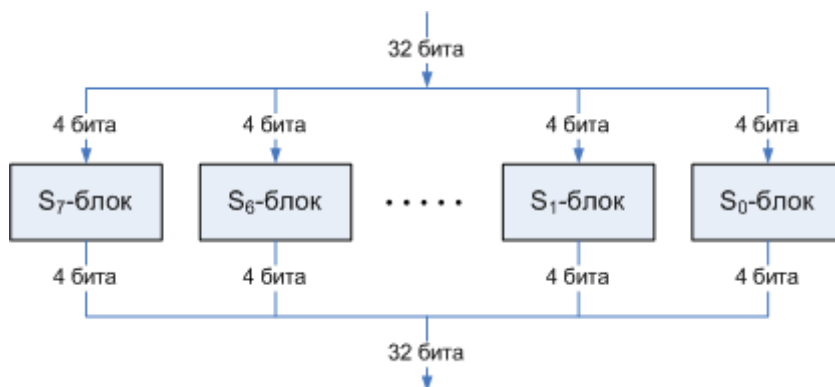


Рис. 5. Подстановка S-блоками

Для этого результат разбивается на восемь 4-битовых кусочка, каждый из которых подается на вход своего S-блока: первые четыре бита в S_0 -блок, вторые – в S_1 -блок и так далее. Каждый S-блок содержит 16 четырехбитовых элемента, нумеруемых с 0 по 15. ГОСТ рекомендует заполнять каждую из восьми таблиц различными числами множества $\{0, 1, 2, \dots, 15\}$, переставленными случайным образом.

S_0 -блок

№	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Элемент	4	10	9	2	13	8	0	14	6	11	1	12	7	15	5	3

S_1 -блок

№	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Элемент	14	11	4	12	6	13	15	10	2	3	8	1	0	7	5	9

S_2 -блок

№	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Элемент	5	8	1	13	10	3	4	2	14	15	12	7	6	0	9	11

S_3 -блок

№	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Элемент	7	13	10	1	0	8	9	15	14	4	6	12	11	2	5	3

S_4 -блок

№	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Элемент	6	12	7	1	5	15	13	8	4	10	9	14	0	3	11	2

S_5 -блок

№	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Элемент	4	11	10	0	7	2	1	13	3	6	8	5	9	12	15	14

S_6 -блок

№	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Элемент	13	11	4	1	3	15	5	9	0	10	14	7	6	8	2	12

S_7 -блок

№	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Элемент	1	15	13	0	5	7	10	4	9	2	3	14	6	11	8	12

Рис. 6. Набор S-блоков Центрального банка РФ

По входным четырем битам определяется номер элемента в S-блоке, который поступает на выход. Выходы всех восьми S-блоков объединяются в 32-битовое слово, затем все слово циклически сдвигается влево на 11 бит. Наконец, результат объединяется с помощью XOR с левой половиной, и получается новая правая половина, а правая половина становится левой половиной. Эти операции выполняются 32 раза. После этого левая и правая части меняются местами.

Запишем базовый цикл алгоритма ГОСТ.

Вход: Блок L, R , раундовый ключ W .

Выход: Преобразованный блок L, R .

FOR $i = 0$ TO 31 DO

$k \leftarrow R \oplus W_i;$ $k = (k_7 \dots k_0)_{16};$ FOR $j = 0$ TO 7 DO $k_j \leftarrow S_j[k_j];$ $L \leftarrow L \oplus (k \ll 11);$ $L \leftrightarrow R;$
--

$L \leftrightarrow R;$

RETURN L, R .

Для шифрования и дешифрования сообщения используется один алгоритм. Единственным различием является генерация раундового ключа. Чтобы дешифровать блок, строим раундовый ключ

$$W = \underbrace{K_0 K_1 K_2 K_3 K_4 K_5 K_6 K_7}_{\text{Раунды 1-8}} \underbrace{K_7 K_6 K_5 K_4 K_3 K_2 K_1 K_0}_{\text{Раунды 9-16}} \underbrace{K_7 K_6 K_5 K_4 K_3 K_2 K_1 K_0}_{\text{Раунды 17-24}} \underbrace{K_7 K_6 K_5 K_4 K_3 K_2 K_1 K_0}_{\text{Раунды 25-32}},$$

подаем на вход Y и на выходе получаем X .

Если исходное сообщение имеет размер больше, чем 64 бита, то оно разбивается на отдельные блоки. Каждый блок шифруется независимо от остальных с использованием одного и того же ключа. Данный режим работы алгоритма называется простой заменой.

Лабораторная работа № 2

«БАЗОВЫЕ АЛГОРИТМЫ ТЕОРИИ ЧИСЕЛ»

Задание. Написать и отладить набор подпрограмм, реализующих базовые алгоритмы, используемые в изученных криптосистемах с открытым ключом: возведение в степень по модулю ($a^x \bmod p$), вычисление наибольшего общего делителя ($\text{НОД}(a, b)$), вычисление инверсии ($x^{-1} \bmod p$).

Общие сведения. Для многих криптографических систем актуален так называемый обобщенный алгоритм Евклида.

Теорема. Пусть a и b – два целых положительных числа. Тогда существуют целые (не обязательно положительные) числа x и y , такие, что

$$ax + by = \text{НОД}(a, b). \quad (1)$$

Обобщенный алгоритм Евклида служит для отыскания наибольшего общего делителя целых чисел a, b ($\text{НОД}(a, b)$) и x, y , удовлетворяющих (1).

Введем три строки $U = (u_1, u_2, u_3)$, $V = (v_1, v_2, v_3)$ и $T = (t_1, t_2, t_3)$. Запишем обобщенный алгоритм Евклида.

Вход: Положительные целые числа $a, b, a \geq b$.

Выход: $\text{НОД}(a, b)$, x, y , удовлетворяющие (1).

$U \leftarrow (a, 1, 0), V \leftarrow (b, 0, 1)$.

WHILE $v_1 \neq 0$ DO

$q \leftarrow u_1 \text{ div } v_1;$
$T \leftarrow (u_1 \bmod v_1, u_2 - qv_2, u_3 - qv_3);$
$U \leftarrow V, V \leftarrow T.$

RETURN $U = (\text{НОД}(a, b), x, y)$.

В алгоритме операция div – это целочисленное деление, mod – остаток от деления.

Пример. Пусть $a = 24, b = 15$. Найдем $\text{НОД}(a, b), x, y$.

				Выход	U	V	
			4 шаг	U	V	T	
		3 шаг	U	V	T		
	2 шаг	U	V	T			
1 шаг	U	V	T				
Значения элементов строк	24	15	9	6	3	0	$q = 2$
	1	0	1	-1	2	-5	$q = 1$
	0	1	-1	2	-3	8	$q = 1$

Вначале в строку U записываются числа $(24, 1, 0)$, а в строку V – числа $(15, 0, 1)$. Вычисляется строка T . После этого в строку U заносятся данные строки V , а в V – данные строки T . Таким образом, на втором шаге цикла WHILE

$U = (15, 0, 1)$, $V = (9, 1, -1)$ и опять вычисляется строка T . Этот процесс продолжается до тех пор, пока первый элемент строки V не станет равным нулю. Тогда строка U (предпоследний столбец в схеме) содержит ответ. В нашем случае $U = (3, 2, -3)$. Выполним проверку $24 \cdot 2 + 15 \cdot (-3) = 3$.

У алгоритма Евклида есть одно важное применение. В некоторых задачах криптографии для заданных чисел e и z требуется найти такое число $d < z$, что

$$ed \bmod z = 1. \quad (2)$$

Такое число d существует тогда и только тогда, когда числа e и z взаимно простые.

Определение. Число d , удовлетворяющее (2), называется инверсией e по модулю z (обозначается $d = e^{-1} \bmod z$).

Равенство (2) означает, что для некоторого целого k

$$ed - kz = 1. \quad (3)$$

Учитывая, что e и z взаимно простые, перепишем (3) в виде

$$z(-k) + ed = \text{НОД}(z, e), \quad (4)$$

что полностью соответствует (1). Поэтому, чтобы вычислить d , нужно просто использовать обобщенный алгоритм Евклида для решения уравнения (4). Значение переменной k нас не интересует, поэтому можно не вычислять вторые элементы строк U , V , T . Если число d получается отрицательным, то нужно прибавить к нему z , так как по определению число берется из множества $\{0, 1, \dots, z-1\}$.

Следующей важной операцией в криптографии с открытыми ключами является операция возведения в степень по модулю. Рассмотрим алгоритм, возвращающий вычисленное значение $a^x \bmod p$, где a , x , p целые числа.

Вход: Целые числа a , $x = (x_t x_{t-1} \dots x_0)_2$, p .

Выход: Число $y = a^x \bmod p$.

$y \leftarrow 1$, $s \leftarrow a$.

FOR $i = 0, 1, \dots, t$ DO

| IF $x_i = 1$ THEN $y \leftarrow y \cdot s \bmod p$;
 | $s \leftarrow s \cdot s \bmod p$.

RETURN y .

В данном алгоритме биты показателя степени просматриваются справа-налево (от младшего бита к старшему), поэтому он называется возведение в степень справа-налево.

Лабораторная работа № 3 «АЛГОРИТМЫ ШИФРОВАНИЯ С ОТКРЫТЫМ КЛЮЧОМ»

Задание. Написать и отладить программу, реализующую алгоритм шифрования RSA для передачи секретных сообщений в адрес абонента В. Рекомендуемые значения параметров $p_B=113$, $q_B=281$, $e_B=3$. В работе использовать подпрограммы Лабораторной работы № 2.

Требование к содержанию отчета. В отчете для примера выполнения программы привести исходные данные: p_B , q_B , e_B , открытый текст m ; расчетные данные: n_B , z_B , d_B , полученный шифротекст c и расшифрованный открытый текст m' .

Общие сведения. Алгоритм шифрования RSA является алгоритмом с открытым ключом. Для генерации двух ключей (личного K_B^- и открытого K_B^+) абоненту В необходимо выполнить следующие действия.

1. Выбрать два больших случайных простых числа p_B и q_B .

2. Вычислить

$$n_B = p_B \times q_B, \quad z_B = (p_B - 1) \times (q_B - 1).$$

3. Выбрать случайным образом простое число e_B , меньшее, чем n_B , у которого нет общих делителей (кроме 1) с числом z_B (взаимно простые числа). Числа e_B и n_B составляют открытый ключ абонента В: $K_B^+ = (e_B, n_B)$.

4. С помощью обобщенного алгоритма Евклида вычислить число d_B ($d_B = e_B^{-1} \bmod z_B$ – инверсия e_B по модулю z_B), такое что остаток от деления $e_B \times d_B$ на z_B был равен 1: $e_B \times d_B \bmod z_B = 1$. Числа d_B и n_B составляют личный ключ абонента В: $K_B^- = (d_B, n_B)$.

Абонент А шифрует сообщение $m < n_B$ по формуле

$$c = m^{e_B} \bmod n_B$$

и пересылает шифротекст c участнику В по открытой линии.

Абонент В, получивший зашифрованное сообщение, вычисляет открытый текст по формуле

$$m' = c^{d_B} \bmod n_B.$$

Для шифрования большого сообщения оно разбивается на маленькие блоки $m_i < n_B$.

Список литературы

1. Мельников, В.П. Информационная безопасность : Учеб. пособие / под ред. С. А. Клейменова. - 8-е изд., испр. - М. : Академия, 2013. - 336 с. - ISBN 978-5-7695-9954-5 : 797-00.
2. Чопоров О.Н. Защита информации и информационная безопасность [Электронный ресурс] : Учеб. пособие. - Электрон. текстовые, граф. дан. (1,8 Мб). - Воронеж : ФГБОУ ВПО "Воронежский государственный технический университет", 2012.
3. Шаньгин, В.Ф. Информационная безопасность и защита информации [Электронный ресурс] : учебное пособие / В.Ф. Шаньгин. - Информационная безопасность и защита информации ; 2019-04-19. -Саратов : Профобразование, 2017. - 702 с. - ISBN 978-5-4488-0070-2. URL: <http://www.iprbookshop.ru/63594.html>
4. Прохорова, О. В. Информационная безопасность и защита информации: учебник / О.В. Прохорова. – Самара : Самарский государственный архитектурно-строительный университет, 2014. - 113 с. –ISBN 978-5-9585-0603-3. URL: <http://biblioclub.ru/index.php?page=book&id=438331>
5. Голиков, А.М. Защита информации от утечки по техническим каналам [Электронный ресурс] : учебное пособие / А.М. Голиков. - Томск : Томский государственный университет систем управления и радиоэлектроники, 2015. - 256 с. URL: <http://www.iprbookshop.ru/72090.html>
6. Гатченко, Н. А. Криптографическая защита информации / Н. А. Гатченко, А. С. Исаев, А. Д. Яковлев. — СПб. : Университет ИТМО, 2012. — 142 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/68658.html>