

# МЕТОД ОБНАРУЖЕНИЯ LSB-ВСТАВОК В ИСКУССТВЕННЫХ ЦВЕТНЫХ ИЗОБРАЖЕНИЯХ С ГРАДИЕНТНОЙ ЗАЛИВКОЙ С НИЗКИМ ЗАПОЛНЕНИЕМ СТЕГОКОНТЕЙНЕРА

В статье представлен метод обнаружения LSB-вставок и области их расположения в искусственных цветных изображениях с градиентной заливкой, эффективный при работе с низким заполнением стегоконтейнера. Метод обнаружения встраивания основывается на анализе комбинаций пикселей нулевого слоя исследуемого изображения и определении уникальных и повторяющихся последовательностей и локализации области встраивания посредством решения задачи о наибольшем пустом прямоугольнике с применением фильтра предварительной обработки полученного массива. Использование предлагаемого метода позволит организациям повысить уровень информационной безопасности за счет своевременного выявления наличия в таких изображениях скрытой злоумышленниками информации и принятия мер по блокировке пересылаемого сообщения.

**Ключевые слова:** стегоанализ, стеганографический анализ, анализ стегоконтейнера, обнаружение LSB-вставки, метод замены наименее значащего бита.

Vilkhovskiy D.E., Guts A.K.

# METHOD OF DETECTING LSB INSERTION IN LOW STEGO-PAYLOAD ARTIFICIAL COLOR IMAGES OF A GRADIENT FILL

The paper presents a method of detecting LSB insertion and its exact location in artificial color images of a gradient fill that is efficient when dealing with low stego-payload. The embedding detection method is based on a binary pixel combination matrix analysis for the zero layer of the image under study in order to determine the presence of unique and repeating sequences, the embedding area locating method that uses an algorithm for largest empty rectangles along with using a preprocessing filter that processes the resulting data array in order to eliminate noise distortions that narrow the area potential embedding area. Using the proposed method enables organizations to enhance their information security by timely detecting the information hidden by intruders in the images and taking measures to block the message from forwarding.

**Keywords:** steganalysis, steganographic analysis, stegocontainer analysis, LSB-insertion detection, least significant bit replacement method.

Популярность и эффективность LSB-вставок, то есть вставок, выполненных методом замены наименее значащего бита, как метода стеганографии объясняется тем, что встраивание в наименее значащий бит не приводит к визуально-заметным артефактам на изображении. При этом, чем меньший уровень заполнения стегоконтейнера, тем сложнее обнаруже-

ние стеговставки, что позволяет злоумышленникам незаметно передавать информацию в пересылаемом медиа-контейнере и ставит под угрозу информационную безопасность организаций.

Работы по стеганографическому анализу, выполненные в течение последних лет, предлагают преимущественно использование сверточных нейронных

сетей [1 –5] или алгоритмов на основе машинного обучения [6 – 8]. При этом применение сверточных нейронных сетей для целей стегоанализа имеет высокую стоимостную нагрузку, обусловленную сложностью вычислений. Использование алгоритмов на основе машинного обучения имеет меньшую стоимостную нагрузку, однако зачастую связано с проблемой выбора ядра и практически не решает задачу определения области встраивания.

Для искусственного изображения с градиентной заливкой применение нейронных сетей или алгоритмов, построенных на машинном обучении экономически нецелесообразно. Следовательно, для таких изображений необходимо разработать такой алгоритм, который бы имел высокую эффективность обнаружения наличия встраивания и высокую точность определения области встраивания при невысокой стоимости расчетов.

Таким образом, целью данного исследования является разработка алгоритма стеганографической атаки на метод LSB замены в искусственных цветных изображениях, позволяющего в автоматическом режиме детектировать скрытое в изображении сообщение. Подобный алгоритм целесообразно интегрировать во внутреннюю систему документооборота организации с целью предотвращения несанкционированной передачи третьим лицам чувствительной информации, т.е. противодействия промышленному шпионажу и повышения информационной, а следовательно, и экономической безопасности организации.

### 1. Общая постановка задачи

Карта битов нулевого слоя пикселей искусственного изображения с градиентной заливкой, в правый нижний угол которого было произведено встраивание наглядно демонстрирует измененную структуру в области встраивания (рис. 1).

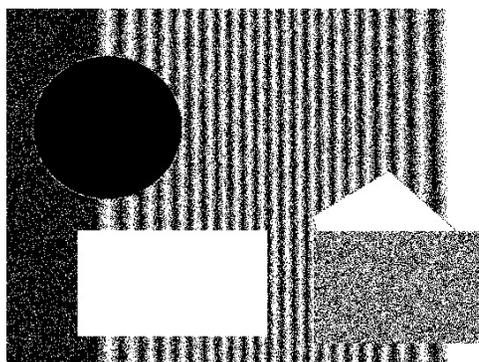


Рис. 1. Карта битов нулевого слоя пикселей искусственного изображения с градиентной заливкой

Анализ множества карт битов нулевого слоя пикселей искусственных изображений с градиентной заливкой, в которые было произведено встраивание методом LSB-замены, позволил установить, что область встраивания содержит преимущественно уни-

кальные комбинации пикселей, тогда как в областях без встраивания в большом количестве присутствуют не уникальные последовательности, формирующие достаточно четко определяемые черно-белые полосы. Пример различий комбинаций пикселей приведен на рисунке 2.



Рис. 2. Различия в комбинациях пикселей на карте битов нулевого слоя в области без встраивания (слева) и со встраиванием (справа)

Автоматическое обнаружение области сосредоточения уникальных комбинаций пикселей позволит выявлять наличие LSB-вставок, а также область их расположения на изображении-стегоконтейнере.

### 2. Характеристика алгоритма обнаружения наличия и месторасположения LSB-вставки на основе анализа комбинаций битов нулевого слоя пикселей

Алгоритм обнаружения встроенного сообщения, выполненного методом LSB-вставки, основывается на последовательном анализе каждого из пикселей нулевого слоя с окном 5 и записи последовательностей каждого окна в бинарном виде.

Окно 5 означает, что анализируется комбинация пикселей с размерностью 5x5, где центральным является анализируемый пиксель. Таким образом, справа, слева, сверху и снизу от исследуемого пикселя располагаются еще по 2 пикселя. Пример двух уникальных комбинаций пикселей с окном 5 приведен на рисунке 3.

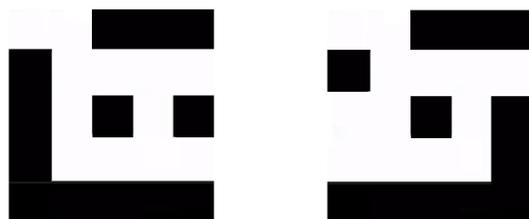


Рис. 3. Пример двух уникальных комбинаций пикселей с окном 5

Так, для окон, представленных на рисунке 3, комбинации имеют следующий вид (табл. 1):

Данные о комбинациях заносятся в хеш-таблицу линейным списком, где ключом является закодированная последовательность окна с исследуемым пик-

Таблица 1

**Комбинации комбинаций пикселей в бинарном виде**

Последовательность для окна а)	Последовательность для окна б)
0 1 1 1 1	1 0 0 0 0
1 0 0 0 1	0 0 0 0 1
0 0 1 0 1	1 0 1 0 1
1 0 0 0 1	0 0 0 0 1
1 1 1 1 1	0 1 1 1 0

селем, а значением – координаты исследуемого (центрального) пикселя.

Кроме того, заполняется еще одна структура, куда заносятся данные о комбинациях, включая саму последовательность и количество ее повторов. В результате, мы получаем бинарную матрицу размером  $N \times M$ , содержащую единичные и нулевые значения, где нулевое значение присваивается пикселю, входящему в уникальную комбинацию, а единичное значение присваивается пикселю, входящему в неуникальную комбинацию. В результате, формируется матрица  $M_0$ , содержащая данные об уникальных и неуникальных комбинациях в бинарном виде.

В графическом виде данные указанной выше структуры представлены в виде массива из черных и белых точек. Пример графического изображения области, содержащей встраивание и области, не содержащей встраивание, в идеальной ситуации представлен на рисунке 4.

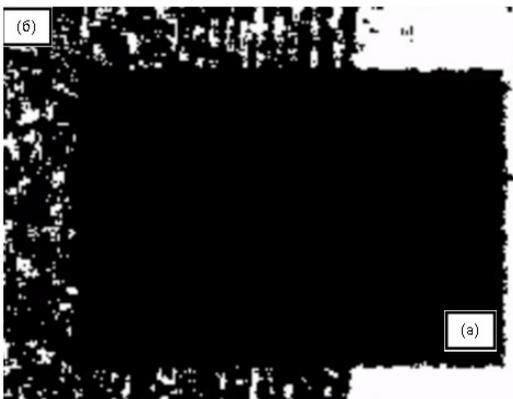


Рис. 4. Участки со встраиванием (а) и без встраивания (б)

Далее переходим к задаче о наибольшем пустом прямоугольнике, решение которой позволит определить наибольший прямоугольник, содержащий только нулевые значения, т.е. только уникальные комбинации.

**3. Задача о наибольшем пустом прямоугольнике**

**3.1. Общая характеристика о наибольшем пустом прямоугольнике**

Задача о наибольшем пустом прямоугольнике впервые была представлена в работах А. Наамад, Д. Ли и В. Шу [9] с последующей корректировкой и уточнениями в работах этих и других авторов [10 – 14].

Решение задачи осуществляется в 2 этапа.

Этап 1. Анализируются отдельные значения элементов исходной бинарной матрицы  $M_0$ . Анализ проводится по рядам с последовательным преобразованием значений каждого из элементов и формированием новой матрицы  $M_1$  по установленной схеме:

1. В строгом порядке все обнаруженные единичные значения элементов матрицы инвертируются в нулевые значения. Инвертирование осуществляется в неизменном виде вне зависимости от того, какое значение (нулевое или единичное) наблюдается у вышестоящих элементов исходной матрицы.

2. Каждому из обнаруженных нулевых значений присваивается порядковый номер его обнаружения по столбцам. Таким образом, всем элементам в первом ряду с нулевыми значениями присваивается значение 1, а элементу во втором ряду с нулевым значением, присваивается значение 2 только при условии, что значение вышестоящего элемента по его столбцу уже равно единице в формируемой матрице, что означает наличие нулевого значения по этому столбцу в исходной матрице. И т.д.

3. Любое инвертирование единичного значения элемента исходной матрицы в нулевое считается разрывом (обнулением) при анализе элементов исходной матрицы с нулевыми значениями. Например, если элемент (3;3) в исходной матрице был инвертирован в 0, то элементу (4;3), имеющему исходное нулевое значение, присваивается значение 1 независимо от того, какие значения присвоены элементам (1;3) и (2;3).

Пример анализа значений элементов бинарной матрицы приведен в таблице 2.

Таблица 2

**Пример анализа значений элементов бинарной матрицы**

Исходная матрица $M_0$						Получаемая матрица $M_1$					
0	0	0	1	0	0	1	1	1	0	1	1
0	0	1	0	0	1	2	2	0	1	2	0
0	1	0	0	0	0	3	0	1	2	3	1
0	0	0	0	0	0	4	1	2	3	4	2
0	0	0	0	0	1	5	2	3	4	5	0
1	0	0	1	0	0	0	3	4	0	6	1

Этап 2. Анализируется матрица  $M_1$  с целью обнаружения прямоугольника с наибольшей площадью:

1) Определяются все прямоугольники, расположенные между нулевыми значениями матрицы  $M_1$ ;

2) Анализ осуществляется последовательно, начиная с самого нижнего ряда;

3) Основание прямоугольника должно лежать на исследуемом ряду;

Ниже приведен пример обнаружения прямоугольника с наибольшей площадью на основе данных, представленных в таблице 1:

$$H[6] = 0\ 3\ 4\ 0\ 6\ 1 \rightarrow 6\ (3 \times 2)$$

$$H[5] = 5\ 2\ 3\ 4\ 5\ 0 \rightarrow 10\ (2 \times 5)$$

$$H[4] = 4\ 1\ 2\ 3\ 4\ 2 \rightarrow 8\ (2 \times 4)$$

$$H[3] = 3\ 0\ 1\ 2\ 3\ 1 \rightarrow 4\ (2 \times 2)$$

$$H[2] = 0\ 3\ 4\ 0\ 6\ 1 \rightarrow 4\ (2 \times 2)$$

$$H[1] = 1\ 1\ 1\ 0\ 1\ 1 \rightarrow 3\ (1 \times 3)$$

Следовательно, искомая область – прямоугольник размером  $(2 \times 5)$ , основанием которого является 5 ряд (рис. 5)

Получаемая матрица $M_1$					
1	1	1	0	1	1
2	2	0	1	2	0
3	0	1	2	3	1
4	1	2	3	4	2
5	2	3	4	5	0
0	3	4	0	6	1

Рис. 5. Определение размеров и расположения наибольшего прямоугольника

### 3.2. Анализ уязвимости алгоритма решения задачи о наибольшем пустом прямоугольнике в рамках стегоанализа

Задача о наибольшем пустом прямоугольнике в качестве ограничителей рассматривает любые препятствия, включая случайные, то есть шумы. В рамках предлагаемого метода стегоанализа, основанного на нахождении уникальных комбинаций пикселей, шумом считается такой пиксель, который был определен как входящий в неуникальную комбинацию, но при этом находится в зоне предполагаемого встраивания, то есть среди множества уникальных комбинаций с высокой степенью плотности.

Выше было указано, что в идеальной ситуации, область со встраиванием должна содержать только

уникальные комбинации пикселей, т.е. графически выражаться в виде прямоугольника, состоящего только из черных пикселей – как представлено на рис. 4. Однако, проведенные нами эмпирические исследования показали, что шумы присутствуют более чем в 99% случаях.

Так, для искусственного изображения с стегоставкой, карта битов которого приведена на рисунке 1, область встраивания содержит шумы, приведенные на рисунке 6.

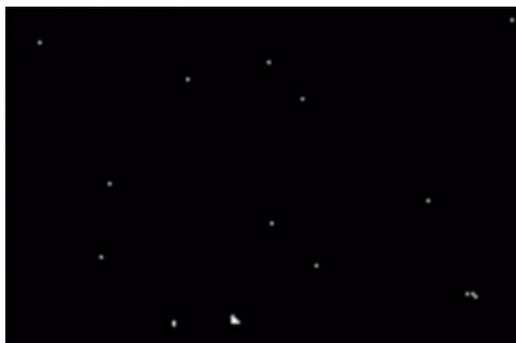


Рис. 6. Карта пикселей с шумом в области со встраиванием

Таким образом, уязвимостью алгоритма решения задачи о наибольшем пустом прямоугольнике в рамках стегоанализа является риск значительного сокращения области предполагаемого встраивания из-за шумовых препятствий. Так, для примера, приведенного на рисунке 6, область обнаружения стегоставки в несколько раз меньше действительной – рисунок 7.



Рис. 7. Область обнаружения стегоставки при наличии шумов

### 3.3. Фильтр предварительной обработки изображения

С целью нивелирования шума и повышением точности обнаружения встраиваемых сообщений, структуру, полученную на 1 этапе решения задачи о наибольшем пустом прямоугольнике, необходимо предварительно обработать при помощи специального фильтра.

Для фильтрации шума последовательно произ-

водится сравнение исследуемого пикселя с двумя соседними пикселями с каждой стороны, а также двумя диагонально прилегающими пикселями. Таким образом, фильтр последовательно анализирует каждый из пикселей структуры с окном 5, в центре которого располагается исследуемый пиксель.

Для максимального учета всех возможных вариаций шумов, предлагается применять коэффициент 0,7. Т.е. если более 70% пикселей исследуемого окна являются пикселями, входящими в уникальную комбинацию (на рисунке – черного цвета), то исследуемый пиксель следует считать случайным шумом, не нарушающим общую картину уникальности имеющихся в этом окне комбинаций. Следовательно, для целей обнаружения стегоконтейнера данный пиксель следует считать пикселем, входящим в уникальную комбинацию, и присвоить ему нулевое значение.

Пример карты пикселей областей со встраиванием и без встраивания до и после обработки фильтром представлены, соответственно, на рисунках 8 и 9.

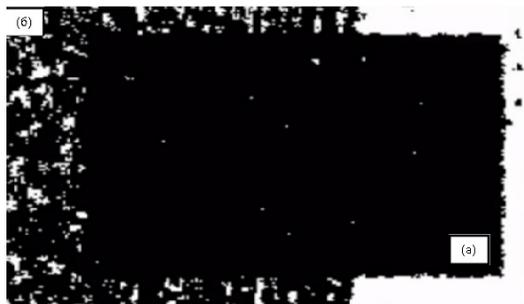


Рис. 8. Карта пикселей областей со встраиванием (а) и области без встраивания (б) до обработки фильтром



Рис. 9. Карта пикселей областей со встраиванием (а) и области без встраивания (б) после обработки фильтром

Таким образом, после обработки фильтром, в области со встраиванием мы получаем однородную карту пикселей, без каких-либо шумов, что позволяет в дальнейшем с максимальной точностью определить область встраивания.

#### Заключение

Алгоритм определяет координаты левого верхнего угла и правого нижнего угла и, основываясь на

найденных координатах, выстраивает обнаруженный прямоугольник. Данный прямоугольник является областью стего-вставки. Решение задачи обнаружения области встраивания с применением данного алгоритма для искусственного изображения, карта пикселей которого представлена на рисунке 1, представлено на рисунке 10, где серый прямоугольник является областью встраивания, обнаруженной при помощи предлагаемого алгоритма.

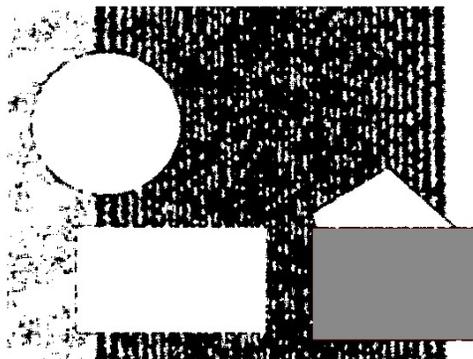


Рис. 10. Карта битов нулевого слоя пикселей искусственного изображения с градиентной заливкой с обнаруженной областью встраивания (серый прямоугольник)

Тестирование предложенного алгоритма обнаружения встраивания, выполненного методом замены наименее значащих бит, было произведено с использованием 1000 цветных искусственных изображений с градиентной заливкой. Встраивание производилось поочередно в красную компоненту (33% всех встраиваний), зеленую компоненту (33% всех встраиваний) и синюю компоненту (34% всех встраиваний). Уровень заполнения стегоконтейнера варьировался от 25 до 10%.

Тестирование показало, что предлагаемый алгоритм способен обнаруживать факт наличия встраивания в 99,2% анализируемых случаях. В среднем, точность обнаружения области встраивания составляет 94,6%, что означает, что границы обнаруженной области встраивания в среднем на 5,4% меньше фактической области встраивания.

Таким образом, можно сделать вывод, что предлагаемый в данной работе метод обнаружения LSB-вставки с низким уровнем заполнения стегоконтейнера обладает высокой эффективностью в искусственных изображениях с градиентной заливкой.

---

## Литература

1. Chaumont M. Deep learning in steganography and steganalysis // In Digital Media Steganography, Academic Press, 2020. P. 321–349.
2. Chen M, Boroumand M, Fridrich J. Deep learning regressors for quantitative steganalysis // Electron Imaging 2018. No 7. P. 160–161.
3. Cogramne R., Giboulot Q., Bas P. The ALASKA steganalysis challenge: A first step towards steganalysis // In Proceedings of the ACM Workshop on Information Hiding and Multimedia Security, 2019. P. 125–137.
4. Xu G., Wu H-Z., and Shi YQ. Ensemble of CNNs for steganalysis: An empirical study // Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security, 2016. P. 103–107.
5. Ye J, Ni J, Yi Y. Deep learning hierarchical representations for image steganalysis // IEEE Trans Inf Forensics Secur. 2017, No 12(11). P. 2545–255.
6. Chaeikar A. Ensemble SW image steganalysis: A low dimension method for LSBR detection // Signal Process Image Commun, 2019. No. 70. P. 233–245.
7. Chaeikar. S.S., Ahmadi A. SW: A blind LSBR image steganalysis technique // In Proceedings of the 10th International Conference on Computer Modeling and Simulation, Sydney Australia, 8 January 2018. P. 14–18.
8. Kumar U.P., Shankar D.D. Blind Steganalysis for JPEG Image using SVM and SVM-PSO Classifiers // International Journal of Innovative Technology and Exploring Engineering (IJITEE). Vol 8. 2019. P. 1239 – 1246.
9. Naamad A., Lee D. T., Hsu W.-L. On the Maximum Empty Rectangle Problem // Discrete Applied Mathematics, 1984. P. 267–277.
10. Acharyya A, De M., Subhas C., Pandit S. Variations of largest rectangle recognition amidst a bichromatic point set // Discrete Applied Mathematics. Vol 286. 2020, P. 35-50.
11. Alok A., Subhash S. Fast algorithms for computing the largest empty rectangle // Proc. 3rd Annu. Symposium on Computational Geometry, 1987. P. 278–290.
12. Chazelle B., Drysdale R. L., Lee D. T. Computing the largest empty rectangle // STACS. Vol. 166. 1984. P. 43–54.
13. Sarkar, A., Biswas, A., Dutt, M., Bhattacharya, A. Finding a largest rectangle inside a digital object and rectangularization // Journal of Computer and System Sciences. Vol 95. 2018. P. 204-217.
14. Subhas C. Nardy, Bhargab B. Bhattacharya. Location of Largest Empty Rectangle among Arbitrary Obstacles // Foundations of Software Technology and Theoretical Computer Science. Vol.880. 1994. P.10-28.

## References

1. Chaumont M. Deep learning in steganography and steganalysis // In Digital Media Steganography, Academic Press, 2020. P. 321–349.
2. Chen M, Boroumand M, Fridrich J. Deep learning regressors for quantitative steganalysis // Electron Imaging 2018. No 7. P. 160–161.
3. Cogramne R., Giboulot Q., Bas P. The ALASKA steganalysis challenge: A first step towards steganalysis // In Proceedings of the ACM Workshop on Information Hiding and Multimedia Security, 2019. P. 125–137.
4. Xu G., Wu H-Z., and Shi YQ. Ensemble of CNNs for steganalysis: An empirical study // Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security, 2016. P. 103–107.
5. Ye J, Ni J, Yi Y. Deep learning hierarchical representations for image steganalysis // IEEE Trans Inf Forensics Secur. 2017, No 12(11). P. 2545–255.
6. Chaeikar A. Ensemble SW image steganalysis: A low dimension method for LSBR detection // Signal Process Image Commun, 2019. No. 70. P. 233–245.
7. Chaeikar. S.S., Ahmadi A. SW: A blind LSBR image steganalysis technique // In Proceedings of the 10th International Conference on Computer Modeling and Simulation, Sydney Australia, 8 January 2018. P. 14–18.
8. Kumar U.P., Shankar D.D. Blind Steganalysis for JPEG Image using SVM and SVM-PSO Classifiers // International Journal of Innovative Technology and Exploring Engineering (IJITEE). Vol 8. 2019. P. 1239 – 1246.
9. Naamad A., Lee D. T., Hsu W.-L. On the Maximum Empty Rectangle Problem // Discrete Applied Mathematics, 1984. P. 267–277.
10. Acharyya A, De M., Subhas C., Pandit S. Variations of largest rectangle recognition amidst a bichromatic point set // Discrete Applied Mathematics. Vol 286. 2020, P. 35-50.

11. Alok A., Subhash S. Fast algorithms for computing the largest empty rectangle // Proc. 3rd Annu. Symposium on Computational Geometry, 1987. P.278–290.
  12. Chazelle B., Drysdale R. L., Lee D. T. Computing the largest empty rectangle // STACS. Vol. 166. 1984. P. 43–54.
  13. Sarkar, A., Biswas, A., Dutt, M., Bhattacharya, A. Finding a largest rectangle inside a digital object and rectangularization // Journal of Computer and System Sciences. Vol 95. 2018. P. 204-217.
  14. Subhas C. Nardy, Bhargab B. Bhattacharya. Location of Largest Empty Rectangle among Arbitrary Obstacles // Foundations of Software Technology and Theoretical Computer Science. Vol.880. 1994. P.10-28.
- 

**ВИЛЬХОВСКИЙ Данил Эдуардович**, аспирант, ассистент кафедры информационной безопасности, Омский государственный университет им. Ф.М. Достоевского. 644077, г. Омск, пр. Мира, 55. E-mail: vilkhovskiy@gmail.com

**ГУЦ Александр Константинович**, доктор физико-математических наук, профессор кафедры кибернетики, Омский государственный университет им. Ф.М. Достоевского. 644077, г. Омск, пр. Мира, 55. E-mail: aguts@mail.ru

**VILKHOVSKY Danil**, postgraduate student, assistant professor of the Department of Information Security, Dostoevsky Omsk State University. 644077, Omsk, Mira Ave., 55. E-mail: vilkhovskiy@gmail.com

**GUTS Alexander**, Doctor of Physical and Mathematical Sciences, professor, of the Department of Cybernetics, Dostoevsky Omsk State University. 644077, Omsk, Mira Ave., 55. E-mail: aguts@mail.ru