ИССЛЕДОВАНИЕ ТОЧЕЧНОГО КАНАЛА СВЯЗИ С ЛОКОМОТИВОМ

Рожнев А.Ю.
Уральский государственный университет путей сообщения, г. Екатеринбург
alexon@k66.ru

Ключевые слова: БЧХ-коды, биты CRC, хэш-функция

Точечный канал связи с локомотивом (ТКС-Л) предназначен для передачи сигналов локомотивной сигнализации от стационарных устройств СЖАТ к локомотивным устройствам безопасности. Система ТКС-Л состоит из стационарного и локомотивного комплектов. В состав стационарного комплекта входят путевые приемоответчики (ПП) и устройства сопряжения с аппаратурой СЖАТ (УСО-ЖАТ). В состав локомотивного комплекта входят локомотивный считыватель (СПП) и устройство сопряжения с локомотивным устройством безопасности (УСО-ЛС).

Коды Боуза–Чоудхури–Хоквингема (БЧХ) – это широкий класс циклических кодов, применяемых для защиты информации от ошибок[4]. Особенностью является построение кода с заранее определёнными корректирующими свойствами – минимальным кодовым расстоянием. При декодировании принятого слова $\tilde{x}$ в качестве решения декодера принимается слово x, в шар радиуса $t = \dfrac{d-1}{2}$ от которого попадает принятое слово, то есть

$$d_H(x,\tilde{x}) \le \left[\frac{d-1}{2}\right], \quad (1)$$

где $d_H(\bullet,\bullet)$ – расстояние в метрике Хэмминга.

Если число обнаруженных ошибок больше $t = \dfrac{d-1}{2}$, то декодер выносит решение о невозможности исправить слово; используются 127-битные и 255- битные слова (европейский стандарт передачи данных).

Шар радиуса 10 с центром *x* содержит все слова, отличающиеся от *x* в *i* координатах, где $i \in \{0,1,...,t\}$. Поэтому шар с центром в *x* содержит точно $|B_t| = |B_t(x)| = \sum_{i=0}^{t} \binom{n}{i} = 2,2828 \cdot 10^{14}$ слов.

Вероятность того, что случайная последовательность символов *y* попадет в один из шаров вокруг кодовых слов. равна отношению числа исправляемых слов к общему числу двоичных слов размера *n*:

$$\Pr = \frac{|C||B_t|}{2^n} = \frac{2^k \sum_{i=0}^{t} \binom{n}{i}}{2^n} = \frac{2,2828 \cdot 10^{14}}{2^{127-64}} \quad (2)$$

По результатам *k* = 100000 модельных испытаний для 127-битных слов получены следующие результаты: время моделирования – 4 ч, событий «ложный приём» на выходе декодера БЧХ – *m* = 1.

Оценим согласие полученной при моделировании оценки вероятности $Pr_m = 10^{-5}$ с ранее выведенной гипотезой $Pr = 2,48 \cdot 10^{-5}$. Для этого воспользуемся вторым неравенством Чебышева. В нашем случае величина вероятности $Pr_m = \dfrac{m}{k}$ является случайной величиной с математическим ожиданием (что предполагается в нулевой гипотезе), равным $p = Pr$.

Для любого $\varepsilon > 0$ согласно неравенству Чебышева имеем

(3) $$Pr\left\{\left|\frac{m}{k}-p\right|\geq\varepsilon\right\}\leq\frac{p(1-p)}{\varepsilon^2 k}.$$

Нулевая гипотеза (согласие) принимается, если правая часть неравенства не меньше выбранного уровня значимости α. Используемыми на практике уровнями значимости являются величины α= 0.01, 0.05, 0.1.

В нашем случае $\varepsilon = 1,48\cdot 10^{-5}$ и

(4) $$\frac{p(1-p)}{\varepsilon^2 k}=1,11,$$

что позволяет принять гипотезу о согласии полученных при моделировании результатов с теоретическими выводами.

Вероятность трансформации нулевого кодового слова в слово $\overline{0}+e$ (e - вектор ошибок) из шара радиуса 10 с центром в некотором кодовом слове $\overline{x}$ веса *j* равна

(5) $$Pr\left\{d(\overline{0}+e,\overline{x})=r\right\}=\sum_{v=0}^{r}\binom{j}{v}\binom{n-j}{r-v}p^{j-v}q^{v}p^{r-v}q^{n-j-(r-v)}=\sum_{v=0}^{r}\binom{j}{v}\binom{n-j}{r-v}p^{j+(r-2v)}q^{n-j-(r-2v)}.$$

Окончательно:

(6) $$Pr\left\{\overline{0}+e\ \hat{I}\ B_l(\overline{x})\right\}=Pr\left\{d(\overline{0}+e,\overline{x})=r\right\}=\sum_{r=0}^{l}\sum_{v=0}^{r}\binom{j}{v}\binom{n-j}{r-v}p^{j+(r-2v)}q^{n-j-(r-2v)}.$$

Вероятность отсутствия обнаружения ошибки равна сумме вероятностей трансформации слова веса 0 в слово веса *j* для всех допустимых значений *j:*

(7) $$Pr_m=\sum_{j=0}^{n}n_j\sum_{r=0}^{l}\sum_{v=0}^{r}\binom{j}{v}\binom{n-j}{r-v}p^{j+(r-2v)}q^{n-j-(r-2v)},$$

где n_j – количество кодовых слов веса *j.*

Вероятность обнаружения ошибочной последовательности декодером БЧХ (но не исправления), то есть потери пакета, равна

(8) $$Pr_p=1-Pr\left\{w(e)\leq t\right\}-Pr_m.$$

Вероятность, что вес вектора ошибки не больше исправляющей емкости кода, равна

(9) $$Pr\left\{w(e)\leq t\right\}=\sum_{i=0}^{l}\binom{n}{i}p^{i}q^{n-i}.$$

В блоке контроля проверочных символов CRC подлежат оценке следующие величины:

– вероятность совпадения значения поля контрольной суммы в пакете случайных данных с действительной контрольной суммой принятой последовательности,

– вероятность совпадения контрольной суммы в поврежденном пакете (неверно декодированным БЧХ-декодером).

Используемый метод CRC (CCITT) имеет следующие параметры:

– длина кодового слова n = 64,

– количество информационных символов k = 48,

– минимальное кодовое расстояние d = 4 (код гарантированно обнаруживает 1 ошибку и большинство ошибок кратности 2 и 3),

– порождающий полином $x^{16}+x^{12}+x^{5}+1$,

– начальное значение регистра 0xFFFF,

– XOR с окончательным значением регистра не производится.

Полный CRC код образуется из циклического кода Хэмминга с параметрами (2^m - 1, 2^m - 1 - *m,* 3) удалением одного информационного бита и последующим расширением этого кода через добавление одного бита дополнительной проверки на четность. Полученный код остается циклическим. Все кодовые слова таких кодов имеют четный вес, поэтому CRC-код обнаруживает все ошибки нечетного веса, кроме того, кодовое расстояние увеличивается на 1 (по сравнению

с кодом Хэмминга) и равно 4, поэтому также обнаруживаются все ошибки веса не более 3. В частности, полный код CRC-16 имеет параметры (32767, 32751) (*m*=15).

Всего различных кодовых слов в используемом коде *C* имеется

(10) $$|C| = 2^k = 2^{48}.$$

Вероятность того, что случайная последовательность символов совпадет с одним из кодовых слов, не превышает отношения числа кодовых слов к общему числу двоичных слов размера *n:*

(11) $$Pr \le \frac{|C|}{2^n} = \frac{2^{48}}{2^{64}} \approx 1,53 \cdot 10^{-5}.$$

По результатам *k*=100000 испытаний получено *m*=1 событие, в котором случайное слов длиной в 64 бита не было отбраковано контролем проверочных бит CRC. Таким образом, оценка вероятности по результатам моделирования $Pr_m = 10^{-5}$, что хорошо согласуется с ранее выведенной гипотезой $Pr = 1,53 \cdot 10^{-5}$.

Допустим, что при передаче произошло ошибок больше, чем допускает обнаруживающая способность кода БЧХ, и в результате передаваемое кодовое слово трансформировалось в другое кодовое слово. Это означает, что кодовое слово по решению декодера БЧХ и передаваемое кодовое слово отличаются, по крайней мере, в *d*=21 позиции. Следовательно, вероятность p_l того, что *i*-й символ в кодовом слове по решению БЧХ-декодера отличается от *i*-го символа в передаваемом слове БЧХ не меньше, чем

(12) $$p_l \ge \frac{21}{127}.$$

Тогда вероятность того, что расстояние Хэмминга между извлеченным (из принятого по решению БЧХ-декодера кодового слова) информационным словом (длиной 64 бита) и передаваемым информационным словом равно *d* символов, есть

(13) $$\binom{64}{d} p_l^d (1-p_l)^{64-d}.$$

Пусть $Pr_{CRCerr.}(d)$ - вероятность совпадения контрольной суммы CRC-16 в информационном слове, находящемся на расстоянии *d* от передаваемого информационного слова. Тогда полная вероятность необнаружения ошибки контролем проверочных символов CRC16 равна

(14) $$Pr_{CRCerr.}(p_l) = \sum_{d=2}^{p_l - n} \binom{64}{d} p_l^d (1-p_l)^{64-d} Pr_{CRCerr}(d).$$

Вероятность $Pr_{CRCerr}(d)$ оценена с помощью моделирования. Последовательность шагов моделирования:

- генерируем случайное слово *y* длиной 48 бит,
- вычисляем вектор проверочных символов y_{CRC16} CRC-16 для *y,*
- добавляем к слову (y, y_{CRC16}) шум *e* веса *d,*
- осуществляем контроль проверочных символов CRC16 для переданного зашумленного слова.

Расчет этой вероятности дает значение

(15) $$Pr_{CRCerr}(p_l) \approx 1,9 \cdot 10^{-5}$$

при $p_l = \frac{21}{127}$.

Теперь рассмотрим теоретическую границу для вероятности необнаруженной ошибки контролем проверочных символов CRC-16 (для случая нулевого начального регистра) с длиной

сообщения 48 бит. Такой код будет линейным подкодом кода с параметрами (32767, 32751), порожденного полиномом

$$g(x) = x^{16} + x^{12} + x^5 + 1 = \sum_{i=0}^{r} g_i x^i \ , \tag{16}$$

где r=16 – число проверочных символов.

Проверочный полином такого кода определяется из соотношения

$$h(x) = \frac{x^{2^m - 1} - 1}{g(x)} = \sum_{i=0}^{n-r} h_i x^i \ , \tag{17}$$

где m=15 и n=2^m -1 есть длина полного циклического кода CRC-16.

При вероятности ошибки в одном бите принятого кодового слова CRC-16, равной p_l, вероятность необнаруженной ошибки контролем проверочных символов CRC-16 (с длиной сообщения 48 бит) равна

$$Pr_{CRCerr.}(p_l) = \sum_{i=1}^{n} A_i p_l^i (1 - p_l)^{n-i}, \tag{18}$$

где $\{A_i\}_{i=0}^{n}$ – распределение весов (весовой спектр) рассматриваемого укороченного циклического кода CRC-16. Общего метода для вычисления весового спектра линейных кодов не существует (NP-полная задача), поэтому требуется вычислительный эксперимент. Благодаря тому, что мы рассматриваем вычисление CRC с нулевым начальным регистром, такой код будет линейным, и этой линейностью мы далее воспользуемся для эффективного вычисления спектра.

Для этого нам потребуется тождество Мак-Вильямс. Пусть C – линейный код с параметрами (n,k) и весовым спектром $\{A_i\}_{i=0}^{n}$, а C^n – ортогональный ему (n, n-k) – код с весовым спектром $\{B_i\}_{i=0}^{n}$. Обозначим

$$A(x) = \sum_{i=0}^{n} A_i \cdot x^i, \quad A(x) = \sum_{i=0}^{n} B_i \cdot x^i. \tag{19}$$

Тогда имеется тождество

$$q^k B(x) = (1 + (q-1)x)^n A\left(\frac{1-x}{1+(q-1)x}\right) , \tag{20}$$

где q=2 – характеристика поля, над которым определен код.

Более того, в этих обозначениях выражение (19) перепишется в виде:

$$Pr_{CRCerr}(p_l) = (1 - p_l)^n (A(p_l) - A_0) . \tag{21}$$

В нашем случае размерность пространства информационных символов ортогонального кода равна 64–48 = 16.

Так как спектр кода и ортогонального ему кода однозначно связаны тождеством Мак-Вильямс, то нам достаточно вычислить спектр ортогонального кода. Для размерности 16 это можно сделать очень быстро. Тогда в наших обозначениях, n=64,k=16, и

$$Pr_{CRCerr}(p_l) = (1 - p_l)^n (2^{k-n}(1 + p_l)^n B\left(\frac{1-p_l}{1+p_l}\right) - 1) . \tag{22}$$

Расчет по этой формуле дает значение

$$Pr_{CRCerr}(p_l) \approx 1,6 \cdot 10^{-5} \tag{23}$$

при $p_l = \frac{21}{127}$, что хорошо согласуется с полученным ранее значением.

Рассмотрим помехоустойчивое кодирование с применением кодов Гоппы.

Определение 1 ([1]). Кодом Гоппы с конструктивным расстоянием d называется альтернантный код с конструктивным расстоянием d, у которого обращение частотного шаблона G имеет ширину d. Иначе говоря, обращение частотного шаблона задается многочленом $G(x)$ степени $d-1$, который называется многочленом Гоппы. Кодом Гоппы в узком смысле называется код Гоппы с $2t$ проверочными частотами с локаторами $\alpha^{n-2t+1}, \alpha^{n-2t+2}, ..., \alpha^{0}$.

Теорема 1 ([1]). Вектор c является кодовым для кода Гоппы с многочленом Гоппы $G(x)$ тогда и только тогда, когда

$$\sum_{i=0}^{n-1} c_i [\alpha^{ij} / G(\alpha^{-i})] = 0, \; j = 0, ..., 2t-1. \tag{24}$$

Теорема 2 ([1]). Минимальное расстояние d^* и размерность k кода Гоппы с многочленом Гоппы степени $2t$ удовлетворяют неравенствам $d^* \geq 2t+1, k \geq n - 2tm.$

В частотной области коды Гоппы допускают описание с помощью изображенного на рис. 1 устройства с регистром сдвига.

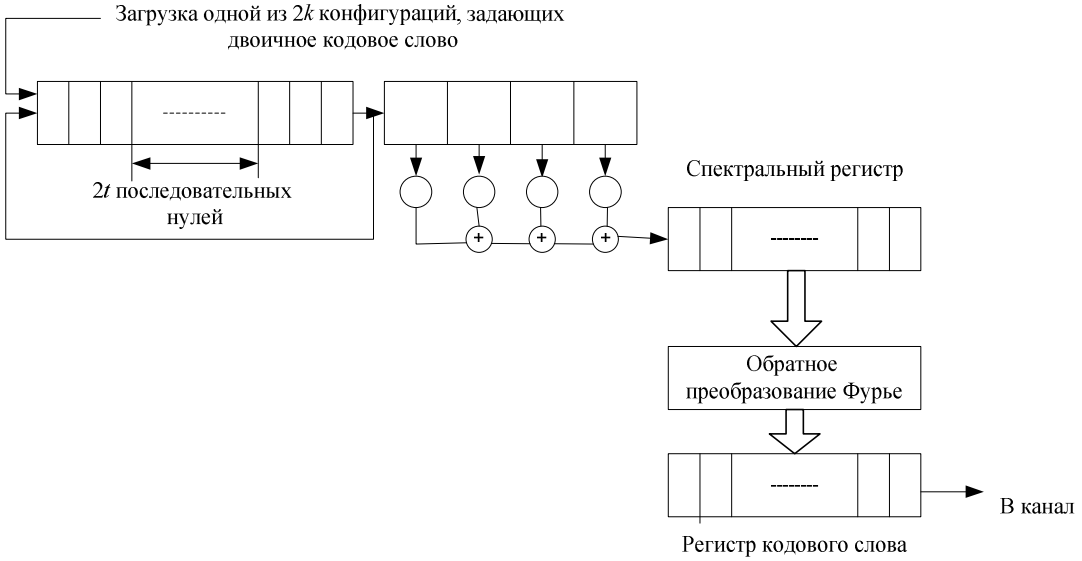

Рис. 1. Частотный кодер для кода Гоппы

Другое описание кодов Гоппы можно дать в следующем виде.

Вместо стандартного введения информации во временную область информация вводится в частотную область – или в спектр или в профильтрованный спектр, как показано на рис. 1, причем в обоих случаях необходимо обеспечить выполнение ограничений сопряженности. Содержащий информацию профильтрованный спектр пропускается через фильтр с конечным импульсным откликом, веса в отводах которого задаются многочленом Гоппы. Фильтр работает циклически: обращение к входу является периодическим. После этого кодовое слово получается обратным преобразованием Фурье спектра.

Теорема 3 ([1]). Код Гоппы в узком смысле над $GF(q)$ длины $n=q^m-1$, задаваемый многочленом Гоппы $G(x)$, состоит из всех векторов $c=(c_0,\ldots,c_{n-1})$ над $GF(q)$, удовлетворяющих условию

$$\sum_{i=0}^{n-1} c_i \prod_{i' \neq i} (x - \alpha^{-i'}) = 0 (\mathrm{mod}\, G(x)). \tag{25}$$

Вернемся теперь к частному случаю двоичных кодов Гоппы, ограничиваясь тем случаем, когда многочлен Гоппы не имеет кратных корней ни в одном расширении данного поля. Такие коды Гоппы называются сепарабельными. Минимальное расстояние сепарабельных двоичных кодов Гоппы равно $2r+1$, где r – степень многочлена Гоппы. Это намного лучше, чем граница для произвольных кодов Гоппы, согласно которой $d^* \geq r+1$. Ключом к доказательству этого

утверждения является следующая теорема, накладывающая ограничения на многочлен, взаимный (возвратный) к локаторному многочлену.

Теорема 4 ([1]). Если определяющий двоичный код Гоппы в узком смысле многочлен Гоппы *G(x)* не имеет корней в поле $GF(2^m)$, то вектор *c* является кодовым словом тогда и только тогда, когда формальная производная $\tilde{\Lambda}'_c(x)$ многочлена, взаимного к локаторному, делится на *G(x).*

Доказательство. Формальная производная многочлена, взаимного к локаторному, равна

$$\tilde{\Lambda}'_c(x) = \sum_{i=1}^{v} \prod_{i' \neq i} (x - \beta_{i'}). \quad (26)$$

Поскольку код является двоичным, то c_i равны нулю или единице, и это выражение для $\tilde{\Lambda}'_c(x)$ следует из формулы, приведенной в теореме 3.

Наименьшим кодом Гоппы является двоичный (8,2,5)-код Гоппы. Выберем $G(x)=x^2+x+1$. Корни этого многочлена различны и лежат в GF(4) и, следовательно, не могут лежать в GF(8). Кодер для такого кода в частотной области изображен на рис. 2. Коды Гоппы очень перспективны.

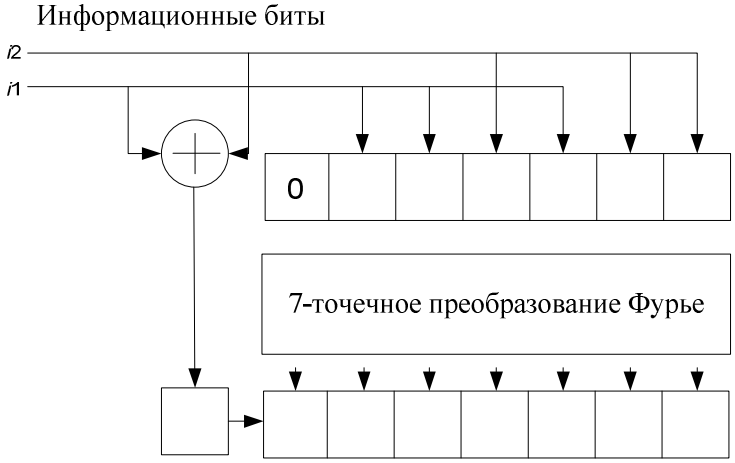

Рис. 2. Схема для (8,2,5)-коды Гоппы

Однонаправленной функцией хэширования или просто *хэш-функцией* называется преобразование информации, переводящее блок информации произвольной длины в блок фиксированной длины [6].

Хэш-функции должны обладать двумя основными свойствами: для данного значения функции *h(M)* невозможно найти аргумент *M;* для данного аргумента *M* должно быть невозможно найти другой аргумент *M'* такой, что *h(M)=h(M').*

Рассмотрим хэш-функции, в которых входной блок информации произвольной длины преобразуется в сжатый 128-битный образ. В каждом цикле хэш-функции MD4 используется своя цикловая функция $f_j, 1 \le j \le 3$, где обозначено

$$\begin{aligned} f_1 &= (X \wedge Y) \vee (\overline{X} \wedge Z); \\ f_2 &= (X \wedge Y) \vee (X \wedge Z) \vee (Y \wedge Z); \quad (27) \\ f_3 &= X \oplus Y \oplus Z. \end{aligned}$$

Схема MD5 сложнее MD4, однако результатом MD5 также является 128-битный хэш-образ. Перейдем к основному циклу алгоритма. Этот цикл продолжается, пока не исчерпаются 512-битные блоки сообщения. Стартовый вектор MD_0 имеет длину 128 бит и представляет собой конкатенацию четырех 32-битных слов $md_0 || md_1 || md_2 || md_3$, где md_0=*01234567,*

md_1=*89abcdef,* md_2=*fedcba98,* md_3=*76543210.* В алгоритме в отличие от MD4 используется четыре цикловые функции f_i, $1 \le j \le 4$, где

$$
\begin{aligned}
f_1 &= (X \wedge Y) \vee (\overline{X} \wedge Z);\\
f_2 &= (X \wedge Z) \vee (X \wedge \overline{Z});\\
f_3 &= X \oplus Y \oplus Z;\\
f_4 &= Y \oplus (X \wedge \overline{Z}).
\end{aligned}
\qquad (28)
$$

В основном цикле MD5 (рис. 3) 16-словного блока M_j, $1 \le j \le n$, обрабатывается за четыре этапа, каждый из которых включает в себя 16 шагов. На каждом шаге *i*-го этапа вычисляется одно из значений $a = b + ((a + f_i(b,c,d) + M_j[s] + t_i) <<< k)$, где $1 \le j \le 4$, $M_j[s]$ - слово, выбранное из *M; s; t_i; k* – параметры шага; *x<<k* – циклический сдвиг значения *x* на *k* бит; «+» – операция сложения по модулю 2^{32}. Причем вычисляемое слово меняется от шага к шагу.

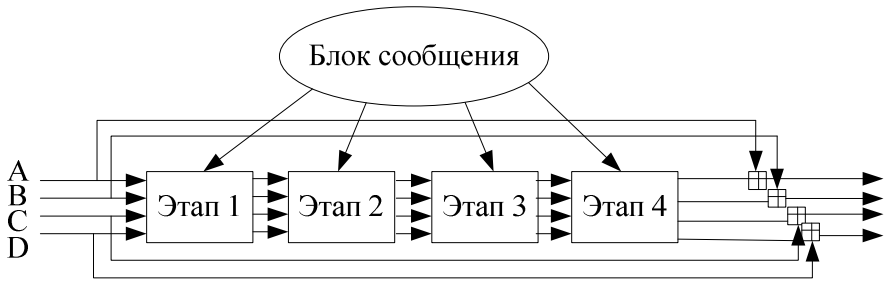

Рис. 3. Схема основного цикла алгоритма MD5

После обработки слова M_n итоговым сжатым образом сообщения будет 128-битная последовательность MD_n=A|B|C|D.

Функция MD5 оказалась стойкой по отношению к дифференциальному криптоанализу каждого из этапов. Комбинированное применение однонаправленных функций совместно с мощными помехоустойчивыми системами кодирования представляется весьма перспективным.

Итак, показана возможность использования помехоустойчивого кодирования сообщений автоматической локомотивной сигнализации, изучены некоторые кодеры, предложено применение кодов Гоппы и функций хэширования для противодействия не только техническим сбоям, но и целенаправленным действиям возможного злоумышленника в целях повышения защищённости передаваемой информации.

Литература

1. *Блейхут Р.* Теория и практика кодов, контролирующих ошибки. - М.: «Мир», 1986 г., С. 25–27.
2. *Казаков А.А.* Автоблокировка, локомотивная сигнализация и автостопы. - М.: «Транспорт», 1980 г., С. 25–50.
3. *Паршин А.В., Субботин Е.А.* Теория передачи сигналов. - Екатеринбург: УрТИСИ ГОУ ВПО «СибГУТИ», 2004 г., С. 20–30.
4. *Росс Н.В.* Элементарное руководство по CRC-алгоритмам обнаружения ошибок. - Rocksoft, 1995 г., 36 с.
5. *Финк Л.М.* Теория передачи дискретных сообщений. - М.: изд-во «Советское радио», 1978 г., 731 с.
6. 6.*Яковлев В.В.* Информационная безопасность и защита информации в корпоративных сетях железнодорожного транспорта. - М.: УМК МПС России, 2002 г., 265 с.