МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
НЕВИННОМЫССКИЙ ТЕХНОЛОГИЧЕСКИЙ ИНСТИТУТ (ФИЛИАЛ)

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ЛАБОРАТОРНЫМ РАБОТАМ

по дисциплине

«Информационная безопасность и защита данных»

Направление подготовки 15.04.04

«Автоматизация технологических процессов и производств»

Направленность (профиль) «Информационно-управляющие системы»

Форма обучения - очно-заочная

Год начала обучения 2022

Реализуется в 5 семестре

Невинномысск 2022

Содержание

Цель и задачи освоения дисциплины (модуля)

Целью является формирование набора профессиональных компетенций будущего магистра по направлению подготовки 15.04.04 Автоматизация технологических процессов и производств.

Задачи изучения дисциплины заключаются в приобретении студентами знаний и практических навыков в области, определяемой основной целью дисциплины.

Наименование компетенций

Код	Формулировка
ПК-9	Способность обеспечивать надежность и безопасность на всех этапах жизненного цикла продукции, выбирать системы экологической безопасности производства

Лабораторная работа № 1

Принципы построения и свойства генераторов псевдослучайных последовательностей

Цель работы: изучить принципы построения и функционирования генераторов псевдослучайных последовательностей для криптографических приложений.

Программа работы

1. Изучить принципы построения и функционирования генераторов псевдослучайных последовательностей для криптографических приложений, основные статистические тесты на случайность данных последовательностей.

2. Разработка программы для реализации заданного генератора псевдослучайной последовательности и оценки ее статистических характеристик.

3. Составить и защитить отчет по результатам работы.

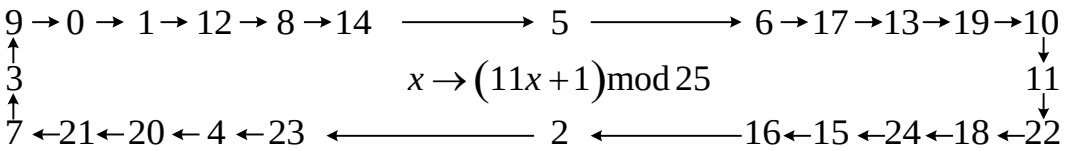
Краткие сведения из теории

Линейные конгруэнтные генераторы (ЛКГ) являются генераторами следующей структуры

$$x_{n+1} = \left(a \cdot x_n + b\right) \bmod m, \qquad (1)$$

где x_{n+1} – (n + 1)-ый член последовательности, x_n – предыдущий член последовательности; константы: a – множитель, b – инкремент, m – модуль.

Период данного генератора не может быть больше m. Если a, b и m выбраны правильно (b обратимо по модулю m; для любого простого p, делящего m, выполняется $a \equiv 1 \left(\bmod p\right)$; если 4 делит m, то $a \equiv 1 \left(\bmod 4\right)$), то генератор будет генератором с максимальным периодом равным m. Например, цикл максимальной длины для ЛКГ по модулю 25:



Преобразование $x \to (3x+1) \bmod 16$ не является циклическим:

Генератор (1) является одношаговым генератором, т.е. имеющий вид $x_{n+1} = f(x_n)$. Данный генератор нельзя использовать в криптографических приложениях, поскольку он предсказуем. Предсказуемыми являются все полиномиальные генераторы, в том числе квадратичные

$$x_{n+1} = \left(ax_n^2 + bx_n + c\right) \bmod m$$

и кубические генераторы

$$x_{n+1} = \left(ax_n^3 + bx_n^2 + cx_n + d\right) \bmod m.$$

ЛКГ сохраняют свою полезность для некриптографических приложений (моделирование, используются в большинстве эмпирических тестов).

Сдвиговые регистры с линейной обратной связью

Сдвиговый регистр с обратной связью состоит из двух частей: сдвигового регистра и функции обратной связи (рисунок 1.1). Сдвиговый регистр представляет собой последовательность битов. Количество битов определяется длиной сдвигового регистра. Если длина равна *n* битам, то регистр называется *n*-битовым сдвиговым регистром.

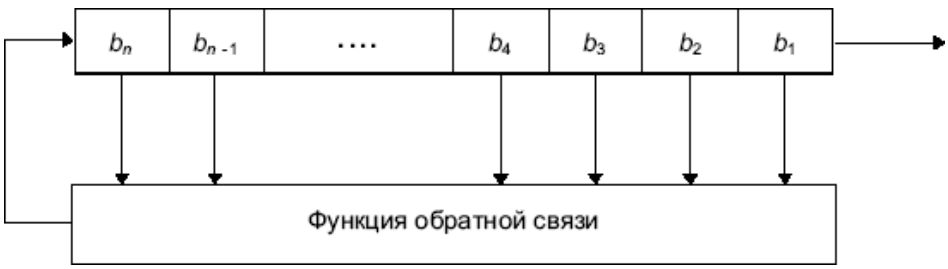


Рисунок 1.1 – Сдвиговый регистр с обратной связью

Всякий раз, когда необходимо извлечь бит, все биты сдвигового регистра сдвигаются вправо на 1 позицию. Новый крайний левый бит является функцией всех остальных битов регистра. На выходе сдвигового регистра оказывается один, обычно младший значащий, бит. Периодом сдвигового регистра называется длина получаемой последовательности до начала ее повторения.

Простейшим видом сдвигового регистра с обратной связью является линейный сдвиговый регистр с обратной связью (linear feedback shift register, LFSR). Обратная связь представляет собой функцию XOR некоторых битов регистра, перечень которых называется отводной последовательностью (рисунок 1.2). LFSR чаще других сдвиговых регистров используется в криптографии.

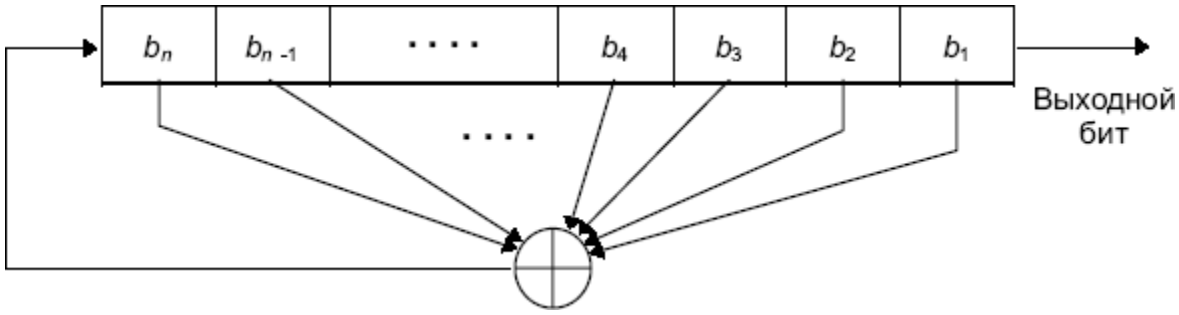


Рисунок 1.2 – Сдвиговый регистр с линейной обратной связью

Например, 4-битовый LFSR с отводом от первого и четвертого битов (рисунок 1.3). Если его проинициализировать значением 1111, то до повторения регистр будет принимать следующие внутренние состояния: 1111, 0111, 1011, 0101, 1010, 1101, 0110, 0011, 1001, 0100, 0010, 0001, 1000, 1100, 1110. Выходной последовательностью будет строка младших значащих битов: 111101011001000.

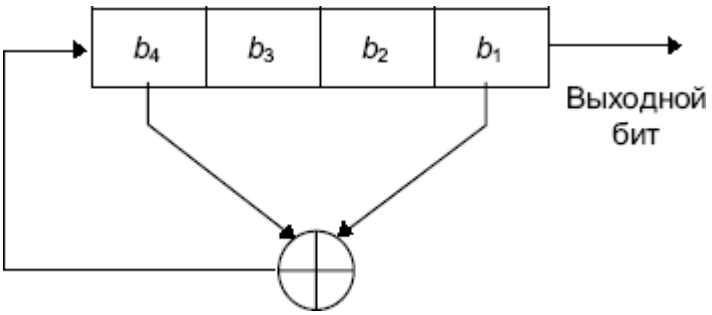


Рисунок 1.3 – 4-битовый LFSR

n-битовый LFSR может находиться в одном из $2^n - 1$ внутренних состояний, т.е. теоретически такой регистр может генерировать последовательность с периодом $2^n - 1$ битов. Только при определенных отводных последовательностях LFSR циклически пройдет через все $2^n - 1$ внутренних состояния (LFSR с максимальным периодом). Для того, чтобы конкретный LFSR имел максимальный период, многочлен, образованный из отводной последовательности и константы 1, должен быть примитивным по модулю 2. Степень многочлена является длиной сдвигового регистра. Примитивный многочлен степени n – это неприводимый многочлен, который является делителем $x^{2^{n-1}} + 1$, но не является делителем $x^d + 1$ для всех d, являющихся делителями $2^n - 1$. Например, примитивный многочлен по модулю 2: $x^{32} + x^7 + x^5 + x^3 + x^2 + x + 1$ или (32, 7, 5, 3, 2, 1, 0), где первое число равно длине LFSR и все числа, за исключением последнего всегда равного 0, задают отводную последовательность, которая отсчитывается от левого края сдвигового регистра. Таким образом, новый бит генерируется с помощью XOR тридцать второго, седьмого, пятого, третьего, второго и первого битов (рисунок 4), циклически проходя до повторения через $2^{32} - 1$ значений.

Код для LFSR(32, 7, 5, 3, 2, 1, 0) на языке C:

```
int LFSR ( ) {
```

```
    static unsigned long ShiftRegister = 1;
    ShiftRegister = ((((ShiftRegister >> 31)
            ^(ShiftRegister >> 6)
            ^(ShiftRegister >> 4)
            ^(ShiftRegister >> 2)
            ^(ShiftRegister >> 1)
            ^ ShiftRegister))
            &0x00000001)
            <<31)
            | (ShiftRegister >> 1);
    return ShiftRegister & 0x00000001;
}
```

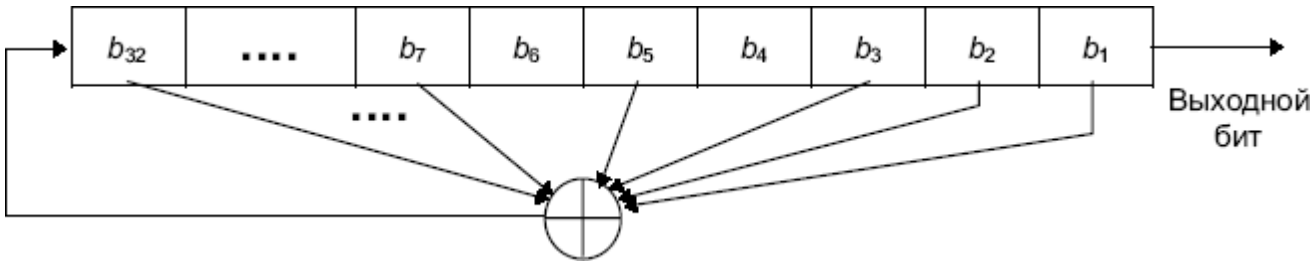


Рисунок 1.4 – 32-битовый LFSR с максимальной длиной

При выборе примитивного многочлена необходимо руководствоваться следующим.

1. Если $p(x)$ примитивен, то примитивен и $x^n p\left(\frac{1}{x}\right)$. Например, если примитивен $(a, b, 0)$, то примитивен и $(a, a - b, 0)$.
2. Быстрее всего программно реализуются примитивные трехчлены (т.е. разряженные многочлены), т.к. для генерации нового бита нужно выполнять XOR только двух битов сдвигового регистра. Однако, разреженность является источником слабости генератора, которой достаточно для вскрытия алгоритма. Для криптографических алгоритмов лучше использовать плотные примитивные многочлены, т.е. у которых много многочленов.

LFSR являются хорошими генераторами псевдослучайных последовательностей, но они обладают нежелательным свойством – последовательные биты линейны, что делает их бесполезными для шифрования. Для LFSR длины n внутреннее состояние представляет собой предыдущие n выходных битов генератора. Поэтому без знания схемы обратной связи она может быть определена по $2n$ выходным битам генератора. Кроме того, большие случайные числа, генерируемые с использованием идущих подряд битов этой последовательности, сильно коррелированны. Несмотря на это LFSR часто используется для создания алгоритмов шифрования.

Статистические тесты псевдослучайных генераторов

Статистические тесты реализуются на основе точно определенных статистик случайной выборки, которые являются функциями элементов случайной выборки (например, число нулей в двоичной последовательности). Статистики выбираются такими, чтобы они были эффективно вычислимыми, и такими, чтобы они соответствовали нормальному или χ^2-распределению.

Предположим, что статистика X случайной последовательности имеет χ^2-ое распределение с v степенями свободы. В таблице 2 представлены значения пороговой величины x_α для заданного уровня значимости α и степени свободы v, такие что выполняется $P\left(X > x_\alpha\right) = \alpha$. Это означает, что если величина X_S статистики выборки выходной последовательности удовлетворяет неравенству $X_S > x_\alpha$, то тест считается проваленным, в противном случае тест считается пройденным.

Если статистика X случайная последовательность, распределенная по нормальному закону $N\left(0,1\right)$, тогда в соответствии с таблицей 1 по за-

данному уровню значимости выбирается пороговая величина x_α, которая удовлетворяет выражению $P(X > x_\alpha) = P(X < -x_\alpha) = \alpha/2$. В этом случае если статистка удовлетворяет неравенству $X_S > x_\alpha$, то тест считается проваленным, в противном случае тест считается пройденным.

Таблица 1 – Процентные точки стандартного нормального распределения

α	0.1	0.05	0.025	0.01	0.005	0.0025	0.001	0.0005
x	1.2816	1.6449	1.9600	2.3263	2.5758	2.8070	3.0902	3.2905

Пусть задана двоичная последовательность $s = s_0, s_1, s_2, ..., s_{n-1}$ длины n. Рассмотрим основные пять статистических тестов:

1. Частотный тест (одноразрядный тест).

В соответствии с этим тестом определяется число нулей n_0 и единиц n_1 в s. Тогда статистика

$$X_1 = \frac{(n_0 - n_1)^2}{n}$$

должна подчиняться χ^2-распределению со степенью свободы 1 и $n \geq 10$.

2. Тест на серии (двухразрядный тест)

В соответствии с тестом определяется числом пар 00, 01, 10, 11 в последовательности s. Определим n_0 и n_1 как количество нулей и единиц в s, соответственно, а через n_{00}, n_{01}, n_{10}, n_{11} определим число пар 00, 01, 10, 11, соответственно. Кроме того $n_{00} + n_{01} + n_{10} + n_{11} = (n-1)$. Используемая статистка в данном тесте

$$X_2 = \frac{4}{n-1}\left(n_{00}^2 + n_{01}^2 + n_{10}^2 + n_{11}^2\right) - \frac{2}{n}\left(n_0^2 + n_1^2\right) + 1$$

подчинена χ^2-распределению со степенью свободы 2 и $n \geq 21$.

3. Обобщенный тест

Пусть m положительное целое такое, что $\left\lfloor \frac{n}{m} \right\rfloor \geq 5 \cdot \left(2^m\right)$, и пусть $k = \left\lfloor \frac{n}{m} \right\rfloor$.

Разделим последовательность s на k непересекающихся частей каждая длиной m, и пусть n_i будет число последовательностей i-ого типа длины m, $1 \leq i \leq 2^m$. Данный тест учитывает последовательности длины m, которые повторяются приблизительное число раз в s. Статистика

$$X_3 = \frac{2^m}{k}\left(\sum_{i=1}^{2^m} n_i^2\right) - k$$

подчинена χ^2-распределению со степенью свободы $2^m - 1$.

4. Тест на последовательности

Данный тест учитывает последовательности произвольной длины ожидаемые в s. Ожидаемое число промежутков (или блоков) длины i в случайной последовательности длины n равно $e_i = \left(n - i + 3\right) \Big/ 2^{i+2}$. Пусть k будет равно наибольшему целому числу i для которой $e_i \geq 5$. Пусть B_i, G_i будет число блоков и промежутков, соответственно, длины i в s для каждой i, $1 \leq i \leq k$. Статистика

$$X_4 = \sum_{i=1}^{k} \frac{\left(B_i - e_i\right)^2}{e_i} + \sum_{i=1}^{k} \frac{\left(G_i - e_i\right)^2}{e_i}$$

подчинена χ^2-распределению со степенью свободы $2k - 2$.

5. Автокорреляционный тест

Целью данного теста является проверка корреляции между последовательностью s и не циклически сдвинутой ее самой. Пусть d будет целое число из интервала $1 \leq d \leq \left\lfloor n/2 \right\rfloor$. Число разрядов в s не равных разрядам в

тех же позициях после сдвига на d равно $A(d) = \sum_{i=0}^{n-d-1} s_i \oplus s_{i+d}$, где $\oplus$ суть оператор XOR.

Таблица 2 – Процентные точки распределения χ^2

	α					
v	0.100	0.050	0.025	0.010	0.005	0.001
1	2.7055	3.8415	5.0239	6.6349	7.8794	10.8276
2	4.6052	5.9915	7.3778	9.2103	10.5966	13.8155
3	6.2514	7.8147	9.3484	11.3449	12.8382	16.2662
4	7.7794	9.4877	11.1433	13.2767	14.8603	18.4668
5	9.2364	11.0705	12.8325	15.0863	16.7496	20.5150
6	10.6446	12.5916	14.4494	16.8119	18.5476	22.4577
7	12.0170	14.0671	16.0128	18.4753	20.2777	24.3219
8	13.3616	15.5073	17.5345	20.0902	21.9550	26.1245
9	14.6837	16.9190	19.0228	21.6660	23.5894	27.8772
10	15.9872	18.3070	20.4832	23.2093	25.1882	29.5883
11	17.2750	19.6751	21.9200	24.7250	26.7568	31.2641
12	18.5493	21.0261	23.3367	26.2170	28.2995	32.9095
13	19.8119	22.3620	24.7356	27.6882	29.8195	34.5282
14	21.0641	23.6848	26.1189	29.1412	31.3193	36.1233
15	22.3071	24.9958	27.4884	30.5779	32.8013	37.6973
16	23.5418	26.2962	28.8454	31.9999	34.2672	39.2524
17	24.7690	27.5871	30.1910	33.4087	35.7185	40.7902
18	25.9894	28.8693	31.5264	34.8053	37.1565	42.3124
19	27.2036	30.1435	32.8523	36.1909	38.5823	43.8202
20	28.4120	31.4104	34.1696	37.5662	39.9968	45.3147
21	29.6151	32.6706	35.4789	38.9322	41.4011	46.7970
22	30.8133	33.9244	36.7807	40.2894	42.7957	48.2679
23	32.0069	35.1725	38.0756	41.6384	44.1813	49.7282
24	33.1962	36.4150	39.3641	42.9798	45.5585	51.1786
25	34.3816	37.6525	40.6465	44.3141	46.9279	52.6197
26	35.5632	38.8851	41.9232	45.6417	48.2899	54.0520
27	36.7412	40.1133	43.1945	46.9629	49.6449	55.4760
28	37.9159	41.3371	44.4608	48.2782	50.9934	56.8923
29	39.0875	42.5570	45.7223	49.5879	52.3356	58.3012
30	40.2560	43.7730	46.9792	50.8922	53.6720	59.7031
31	41.4217	44.9853	48.2319	52.1914	55.0027	61.0983
63	77.7454	82.5287	86.8296	92.0100	95.6493	103.4424
127	147.8048	154.3015	160.0858	166.9874	171.7961	181.9930
255	284.3359	293.2478	301.1250	310.4574	316.9194	330.5197
511	552.3739	564.6961	575.5298	588.2978	597.0978	615.5149
1023	1081.3794	1098.5208	1113.5334	1131.1587	1143.2653	1168.4972

Статистика

$$X_5 = 2\left(A(d) - \frac{n-d}{2} \right) \Big/ \sqrt{n-d}$$

подчинен стандартному нормальному распределению, если $n - d \geq 10$.

Пример. Рассмотрим неслучайную последовательность s длиною $n = 160$, полученную повторением следующей последовательности 4 раза: 11100 01100 01000 10100 11101 11100 10010 01001.

1. Частотный тест. $n_0 = 84$, $n_1 = 76$ и величина статистики $X_1 = 0.4$.

2. Тест на серии. $n_{00} = 44$, $n_{01} = 40$, $n_{10} = 40$, $n_{11} = 35$ и величина статистики $X_2 = 0.6252$.

3. Обобщенный тест. Имеется $m = 3$ и $k = 53$. Блоки 000, 001, 010, 011, 100, 101, 110, 111 встречаются 5, 10, 6, 4, 12, 3, 6 и 7 раз, соответственно, и величина статистики $X_3 = 9.6415$.

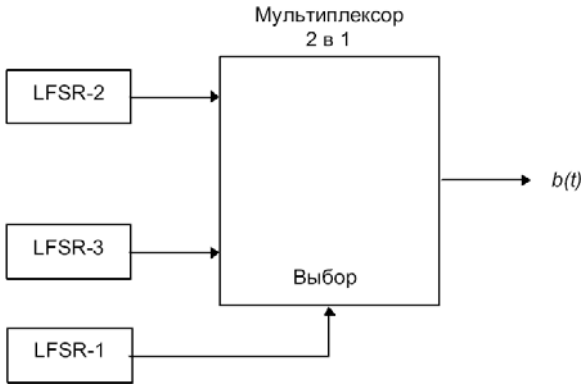
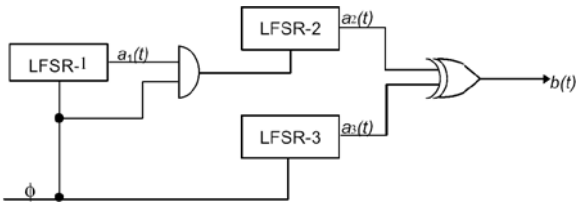
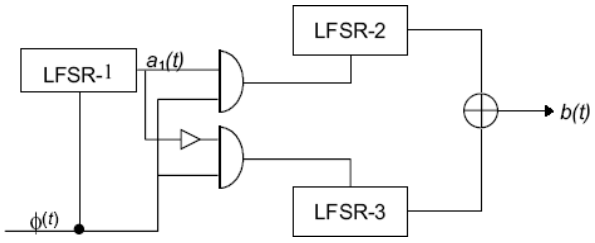
4. Тест на последовательности. $e_1 = 20.25$, $e_2 = 10.0625$, $e_3 = 5$ и $k = 3$. Тогда имеются 25, 4, 5 блоков длины 1, 2, 3, соответственно, и 8, 20, 12 промежутков длины 1, 2, 3, соответственно. Величина статистики $X_4 = 31.7913$.

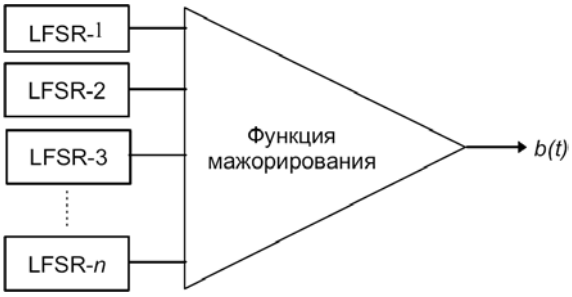
5. Автокорреляционный тест. Если $d = 8$, тогда $A(8) = 100$. Величина статистики $X_5 = 3.8933$.

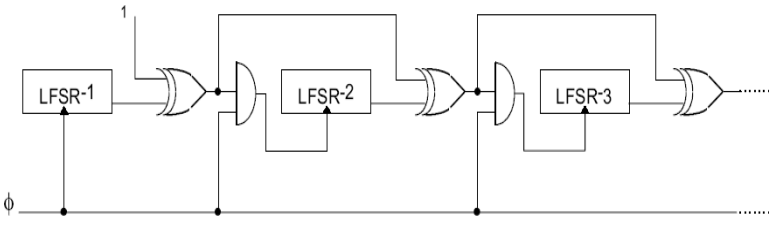
Для уровня значимости $\alpha = 0.05$ пороговые величины для X_1, X_2, X_3, X_4 и X_5 равны 3.8415, 5.9915, 14.0671, 9.4877 и 1.96, соответственно. Как видно заданная последовательность успешно прошла частотный, тест на серии и обобщенный тест, но провалила тест на последовательности и автокорреляционный тест.

Задания на лабораторную работу

1. Написать программу для генератора псевдослучайной последовательности и сгенерировать последовательность равную периоду генератора. Записать полученную последовательность в файл.

Генератор	№ вв	LFSR
Генератор Геффа  $b = (a_1 \wedge a_2) \oplus ((\neg a_1) \wedge a_3)$ Если длины LFSR равны n_1, n_2, n_3, соответственно, то период генератора равен наименьшему общему делителю периодов трех генераторов	1	(16,5,3,2,0) (8,4,3,2,0) (32,7,6,2,0)
	2	(24,4,3,1,0) (20,3,0) (12,6,4,1,0)
	3	(32,7,6,2,0) (28,3,0) (16,5,3,2,0)
Генератор «стоп-пошел» Beth-Piper  	4	(32,7,5,3,2,1,0) (12,6,4,1,0) (28,3,0)
	5	(28,3,0) (16,5,3,2,0) (12,6,4,1,0)
	6	(20,3,0) (28,3,0) (16,5,3,2,0)
Чередующийся генератор «стоп-пошел»  	7	(12,6,4,1,0) (16,5,3,2,0) (24,4,3,1,0)
	8	(8,4,3,2,0) (32,7,6,2,0) (16,5,3,2,0)
	9	(28,3,0) (32,7,5,3,2,1,0) (16,5,3,2,0)

<table>
<tr><td rowspan="3">Пороговый генератор
 

$b = (a_1 \wedge a_2) \oplus (a_1 \wedge a_3) \oplus (a_2 \wedge a_3)$</td><td>10</td><td>n = 3, (32,7,6,2,0)
(28,3,0)
(16,5,3,2,0)</td></tr>
<tr><td>11</td><td>n = 5,
(32,7,5,3,2,1,0)
(16,5,3,2,0)
(24,4,3,1,0)
(8,4,3,2,0)
(20,3,0)</td></tr>
<tr><td>12</td><td>n = 5, (16,5,3,2,0)
(12,6,4,1,0)
(32,7,6,2,0)
(24,4,3,1,0)
(28,3,0)</td></tr>
<tr><td rowspan="3">Каскад Голлманна
 
</td><td>13</td><td>K = 3, (20,3,0)
(24,4,3,1,0)
(28,3,0)</td></tr>
<tr><td>14</td><td>K = 3, (8,4,3,2,0)
(32,7,6,2,0)
(16,5,3,2,0)</td></tr>
<tr><td>15</td><td>K = 3, (32,7,6,2,0)
(28,3,0)
(16,5,3,2,0)</td></tr>
</table>

2. Написать программу для тестирования заданного в генератора по пяти основным тестам.

Контрольные вопросы

1. Что такое LFSR?

2. Как построить псевдослучайный генератор на основе регистра сдвига?

3. Какие тесты на случайность вам известны?

Лабораторная работа №2

Генератор псевдослучайных последовательностей на основе сложностно-теоретического подхода

Цель работы: изучить принципы построения и функционирования генераторов псевдослучайных последовательностей для криптографических приложений, получить навыки разработки программ для работы с длинными числами по модулю простого числа.

Программа работы

1. Изучить принципы построения и функционирования генераторов псевдослучайных последовательностей для криптографических приложений, основные статистические тесты на случайность данных последовательностей.

2. Разработка программы для реализации заданного генератора псевдослучайной последовательности и оценки ее статистических характеристик.

3. Составить и защитить отчет по результатам работы.

Краткие сведения из теории

В данном подходе используется в качестве базиса для криптосистемы некоторая известная и сложная проблема, например, теоретико-числовая: факторизация чисел или нахождение дискретного логарифма. Также как и алгоритмы с открытыми ключами, данные генераторы являются медленными и громоздкими.

RSA генератор

Алгоритм функционирования RSA генератора псевдослучайной последовательности $z_1, z_2, ..., z_l$ длины l:

1. Задаются начальные параметры: генерируется два секретных больших простых числа p и q, вычисляется $n = pq$ и $\phi = (p-1)(q-1)$. Выбирается случайное целое число e из диапазона $1 < e < \phi$, такое что $\gcd(e,\phi) = 1$.

2. Выбирается случайное стартовое целое число x_0 из интервала $[1, n-1]$.

3. Для i от 1 до l выполняется:

3.1. $x_i \leftarrow x_{i-1}^e \bmod n$.

3.2.3$z_i \leftarrow$ младший значащий бит x_i.

.

2

.

4. Выходная последовательность есть $z_1, z_2, ..., z_l$.

Безопасность RSA генератора опирается на сложность вскрытия RSA. Если n достаточно велико (разрядность 1024 бит), то генератор безопасен.

Модификация RSA генератора (Micali-Schnorr RSA генератор):

---

1. Задаются начальные параметры: генерируется два секретных больших простых числа p и q, вычисляется $n = pq$ и $\phi = (p-1)(q-1)$. Принимается $N = \lfloor \log_2 n \rfloor + 1$. Выбирается случайное целое число e из диапазона $1 < e < \phi$, такое что $\gcd(e,\phi) = 1$ и $80e \le N$. Принимается $k = \left\lfloor N\left(1 - \frac{2}{e}\right)\right\rfloor$ и $r = N - k$.

2. Выбирается случайное стартовое целое число x_0 разрядностью r.

3. Генерируется псевдослучайная последовательность длины $k \cdot l$. Для i от 1 до l выполняется:

3.1. $y_i \leftarrow x_{i-1}^e \bmod n$.

3.2.3 .2. $x_i \leftarrow r$ старших значащих

разрядов y_i.

3.3.3$z_i \leftarrow k$ младших значащих разрядов y_i.

.

3

.

4. Выходная последовательность есть $z_1 \| z_2 \| ... \| z_l$, где $\|$ – оператор конкатенации.

Micali-Schnorr RSA генератор более эффективен чем простой RSA генератор, потому что посредством возведения в степень генерируются сразу $\left\lfloor N\left(1-\frac{2}{e}\right)\right\rfloor$ разрядов выходной последовательности. Например, e = 3 и N = 1024, тогда k = 341. Кроме того, каждое возведение в степень включает только одно возведение в квадрат по модулю r = 683-разрядное число и одно модулярное умножение.

Blum-Micali генератор

Безопасность этого генератора определяется трудностью вычисления дискретных логарифмов. Пусть g и p – простые числа. Ключ x_0 начинает процесс:

$$x_i \leftarrow g^{e}_{i-1} \bmod p.$$

Выходом генератора является 1 если $x_i < (p-1)/2$ и 0 в противном случае. Если p достаточно велико, чтобы вычисление дискретных логарифмов $\bmod p$ стало физически невозможным, то этот генератор безопасен.

Blum-Blum-Shub генератор псевдослучайных разрядов

Генератор с квадратичным остатком (BBS генератор) основывается на сложности факторизации числа.

Алгоритм BBS генератора:

---

1. Задаются начальные параметры: генерируется два секретных простых числа p и q, каждый из которых конгруэнтен 3 по модулю 4,вычисляется

$n = pq$.

2. Выбирается случайное целое число s из интервала $[1, n-1]$ такое что $\gcd(s, n) = 1$ и вычисляется $x_0 \leftarrow s^2 \bmod n$.

3. Для *i* от 1 до *l* выполняется:

3.1. $x_i \leftarrow x_{i-1}^2 \bmod n$.

3.2.3$z_i \leftarrow$ младший значащий бит *xi*.

.

2

.

4. Выходная последовательность есть $z_1, z_2, ..., z_l$.

Генерация каждого псевдослучайного бита z_i включает одно возведение в квадрат по модулю. Если *n* достаточно велико (разрядность 1024 бит), то генератор безопасен.

Задание на лабораторную работу

1. Написать программу для генераторов и провести их тестирование

№ вв	Генератор	№ вв	
1	RSA генератор p = 1019, q = 1021	8	Blum-Micali генератор p = 103483
2	RSA генератор p = 1031, q = 1033	9	Blum-Micali генератор p = 103591
3	RSA генератор p = 1039, q = 1049	10	Blum-Micali генератор p = 103703
4	Micali-Schnorr RSA генератор p = 1051, q = 1061	11	Blum-Micali генератор p = 104399
5	Micali-Schnorr RSA генератор p = 1063, q = 1069	12	Blum-Blum-Shub генератор p = 1103, q = 1109
6	Micali-Schnorr RSA генератор p = 1087, q = 1091	13	Blum-Blum-Shub генератор p = 1117, q = 1123
7	Micali-Schnorr RSA генератор p = 1093, q = 1097	14	Blum-Blum-Shub генератор p = 1129, q = 1151
		15	Blum-Blum-Shub генератор p = 1153, q = 1163

2. Написать программу для тестирования заданного в генератора

по пяти основным тестам.

Алгоритм вычисления $a^d \bmod m$

Представим d в двоичной системе счисления $d = \sum_{i=0}^{r} d_i \cdot 2^{r-i}$. Положим $a_0 = a$ и затем для i = 1, …, r вычислим $a_i \equiv a_{i-1}^2 \cdot a^{d_i} \left(\bmod m\right)$. a_r есть искомый вычет $a^d \bmod m$.

Алгоритм модульного умножения

```
F = 0
G = 0
for i from n to 0 do
begin
  F = 2*F + A*b_i
  Q = |F/n|
  F = F – Q*n
end
return (F)
```

Алгоритм модулярной редукции через сокращение разрядности

$$A(j+1) = \sum_{i=0}^{k-1} \left|2^i\right|_p^+ \left\{A(j)\right\}^{[i]}, \text{ для j = 0, 1, 2 …,}$$

где $\{\bullet\}^{[i]}$ – оператор извлечения i-го разряда двоичного представления числа; $|\bullet|_p^+$ – вычетная функция, возвращающая наименьший неотрицательный вычет данного числа по модулю p.

Контрольные вопросы

1. На чем базируется стойкость генераторов псевдослучайных чисел, исследованных в лабораторной работе?

2. Как реализовать возведение в степень чисел большой разрядности по большому модулю?

3. Сравните результаты тестов генераторов из первой лабораторной работы с тестами второй работы.

Лабораторная работа № 3

Разработка программного средства для реализации алгоритма блочного шифрования и дешифрования

Цель работы: изучить принципы блочного шифрования, получить навыки разработки программ для реализации блочного шифрования и дешифрования.

Программа работы

1. Изучить режимы шифрования блочных шифров и стандарты блочного шифрования.
2. Разработка программы для реализации блочного шифрования и дешифрования.
3. Составить и защитить отчет по результатам работы.

Краткие сведения из теории

Криптографическое преобразование составляет основу любого блочного шифра. Прямое криптографическое преобразование (шифрование) переводит блок открытого текста в блок шифротекста той же длины. Обратное криптографическое преобразование (дешифрование) переводит блок шифротекста в исходный блок открытого текста. Необходимое условие выполнения как прямого, так и обратного криптографического преобразования – наличие секретного ключа. Шифры, в которых прямое и обратное преобразования выполняются над блоками фиксированной длины, называются блочными. Для многих блочных шифров разрядность блока составляет 64 бита. Прямое криптографическое преобразование обладает следующим свойством: различные блоки открытого текста отображаются в различные блоки шифротекста. При обратном преобразовании соответствие сохраняется. Прямое преобразование можно рассматривать как перестановку на множестве сообще-

ний с фиксированным размером блока. Результат перестановки носит секретный характер, что обеспечивается секретным компонентом – ключом.

Принцип итерирования является основным при разработке криптографических преобразований и заключается в многократной, состоящей из нескольких циклов обработке одного блока открытого текста. На каждом цикле данные подвергаются специальному преобразованию при участии вспомогательного ключа, полученного из заданного секретного ключа. Выбор числа циклов определяется требованиями криптостойкости и эффективности реализации блочного шифра. Как правило, чем больше циклов, тем выше криптостойкость и ниже эффективность реализации (больше задержка при шифровании/дешифровании) блочного шифра, и наоборот. Так, например, в случае DES (федеральный криптостандарт США) для того, чтобы все биты шифротекста зависели от всех битов ключа и всех битов открытого текста, необходимо 5 циклов криптографического преобразования. DES с 16 циклами обладает высокой криптостойкостью по отношению к ряду криптоаналитических атак.

Конструкция Фейстеля

Конструкция Фейстеля (H. Feistel), или сеть Фейстеля, представляет собой разновидность итерированного блочного шифра. При шифровании блок открытого текста разбивается на две равные части – правую и левую. Очевидно, что длина блока при этом должна быть четной. На каждом цикле одна из частей подвергается преобразованию при помощи функции f и вспомогательного ключа k_i, полученного из исходного секретного ключа. Результат операции суммируется по модулю 2 (операция XOR) с другой частью. Затем левая и правая части меняются местами. Схема конструкции Фейстеля представлена на рисунке 1. Преобразования

на каждом цикле идентичны, но на последнем не выполняется перестановка. Процедура дешифрования аналогична процедуре шифрования, однако k_i, выбираются в обратном порядке. Конструкция Фейстеля хороша тем, что прямое и обратное криптографические преобразования для такого блочного шифра имеют идентичную структуру.

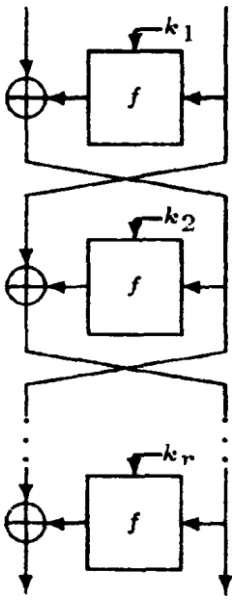


Рисунок 3.1 – Схема конструкции Фейстеля

Конструкция Фейстеля применяется в криптоалгоритмах DES, ГОСТ 28147-89, Lucifer, FEAL, Khufu, Khafre, LOKI, COST, CAST, Blowfish, и др. Блочный шифр, использующий такую конструкцию, является обратимым и гарантирует возможность восстановления входных данных функции f на каждом цикле. Сама функция f не обязательно должна быть обратимой. При задании произвольной функции f не потребуется реализовывать две различные процедуры – одну для шифрования, а другую для дешифрования. Структура сети Фейстеля автоматически позаботится об этом.

Идею конструкции Фейстеля можно объяснить с помощью инволютивного отображения. Так, некоторая функция f является инволюцией,

если $f\left(f\left(x\right)\right)=x$ для всех x. Для такой функции область определения (множество аргументов x) и область значений (множество значений $f\left(x\right)$) совпадают. Например, функция $f\left(x\right)=\neg x$ является инволюцией, так как $f\left(f\left(x\right)\right)=f\left(\neg x\right)=\neg\left(\neg x\right)=x$. Другой пример инволюции: $f\left(x\right)=x\oplus c$, где c — некоторая константа. Действительно, $f\left(f\left(x\right)\right)=f\left(x\oplus c\right)=x\oplus c\oplus c=x$.

Режимы шифрования блочных шифров

При использовании блочных шифров применяются различные схемы шифрования, известные под названием рабочих режимов шифрования для блочных шифров. Очевидно, что применение того или иного режима шифрования не должно отрицательно сказываться на эффективности и тем более криптостойкости блочного шифра. Режимы шифрования позволяют реализовать дополнительные, отсутствующие в исходной конструкции блочного шифра функции.

Стандарт режимов шифрования для блочных шифров (применительно к криптоалгоритму DES) опубликован в материалах Национального института стандартов США и ANSI X3.106. Стандарт включает шифрование в следующих режимах: Электронной кодовой книги (Electronic Code Book, ECB), Сцепления блоков шифра (Cipher Block Chaining, CBC), Обратной связи по шифротексту (Cipher Feedback, CFB) и Обратной связи по выходу (Output Feedback, OFB). В режиме ECB (рисунок 2) шифрование/дешифрование i-го блока открытого текста/шифротекста выполняется независимо: $m_i=D_k\left(c_i\right),\ c_i=E_k\left(m_i\right)$, где через E_k и D_k обозначены процедуры шифрования/дешифрования на секретном ключе k.

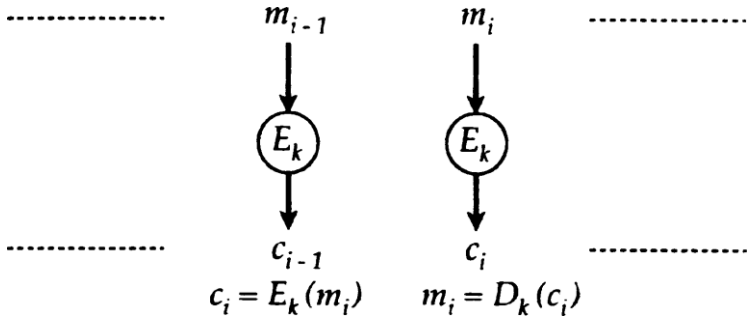


Рисунок 3.2 – Шифрование в режиме ECB

Криптостойкость режима ECB не ниже, чем криптостойкость используемого блочного шифра. Недостаток заключается в том, что фиксированные блоки открытого текста (например, последовательность нулей длины $l = nb$ бит, где b длина блока) будут соответствовать фиксированным блокам шифротекста. Следовательно, открытый текст может быть легко. Скорость обработки блоков в режиме ECB фиксирована и определяется эффективностью реализации блочного шифра. Режим ECB допускает эффективное распараллеливание вычислений. Однако конвейерная обработка блоков в данном режиме невозможна.

В режиме CBC каждый i-й блок открытого текста суммируется по модулю 2 (операция XOR) с $(i – 1)$-м блоком шифротекста и затем шифруется (рисунок 3). Начальное значение задается вектором инициализации.

Криптостойкость режима CBC определяется криптостойкостью используемого блочного шифра. Применение режима CBC позволяет устранить недостаток режима ECB: каждый блок открытого текста «маскируется» блоком шифротекста, полученным на предыдущем этапе. Таким образом, возможность изменения открытого текста при использовании режима CBC весьма ограничена – любые манипуляции с блоками шифротекста, за исключением удаления первого и последнего блоков,

будут обнаружены. Скорость обработки в данном режиме не ниже производительности блочного шифра – задержка при выполнении операции XOR пренебрежимо мала. Процедура шифрования в режиме CBC с трудом поддается распараллеливанию, процедуру дешифрования распараллелить значительно проще.

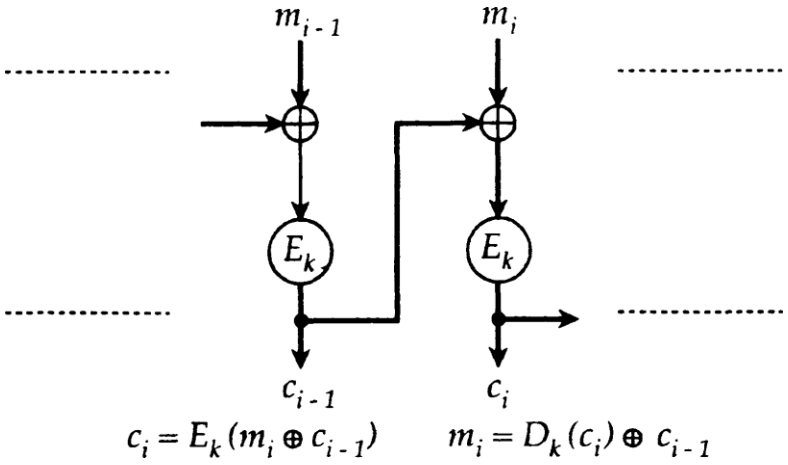


Рисунок 3.3 – Шифрование в режиме CBC

В режиме CFB i-й блок шифротекста формируется путем шифрования $(i - 1)$-го блока шифротекста и его суммированием (операция XOR) с i-м блоком открытого текста (рисунок 4).

Режим CFB можно задать таким образом, что обратная связь будет захватывать не целый n-битный блок, а только k бит предыдущего блока, $k < n$. Начальное значение c_0 так же, как в режиме CBC, задается при помощи вектора инициализации.

Криптостойкость CFB определяется криптостойкостью используемого шифра. Фиксированные блоки открытого текста «маскируются» блоками шифротекста. Возможности изменения открытого текста те же, что и в режиме CBC. Если в режиме CFB с полноблочной обратной связью имеется два идентичных блока шифротекста, результат, например, DES-шифрования на следующем шаге будет тем же. Скорость шифрова-

ния CFB-режима с полноблочной обратной связью та же, что и у блочного шифра, причем возможности распараллеливания процедуры шифрования ограничены.

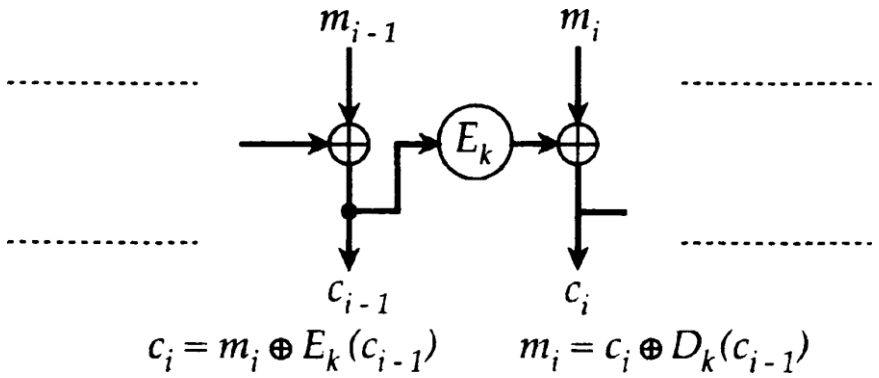


Рисунок 3.4 – Шифрование режима CFB

Режим OFB аналогичен CFB, за исключением того, что суммируемые с открытым текстом биты генерируются независимо от открытого текста и шифротекста. Вектор инициализации s_0 задает начальное значение последовательности блоков s_i, и каждый блок s_i получается путем шифрования предыдущего блока s_{i-1}. Открытый текст шифруется суммированием (операция XOR) i-го блока открытого текста с s_i из независимой последовательности блоков (рисунок 5).

Обратная связь по выходу на k разрядов не рекомендуется из соображений криптостойкости. Режим OFB имеет следующее преимущество по сравнению с режимом CFB: ошибки, возникающие в результате передачи по каналу с шумом, при дешифровании не «размазываются» по всему шифротексту, а локализуются в пределах одного блока. Однако открытый текст может быть изменен путем определенных манипуляций с блоками шифротекста. Скорость шифрования в режиме OFB та же, что и у блочного шифра. Несмотря на то, что OFB-шифрование не поддается распараллеливанию, эффективность процедуры может быть повышена за

счет предварительной генерации независимой последовательности блоков.

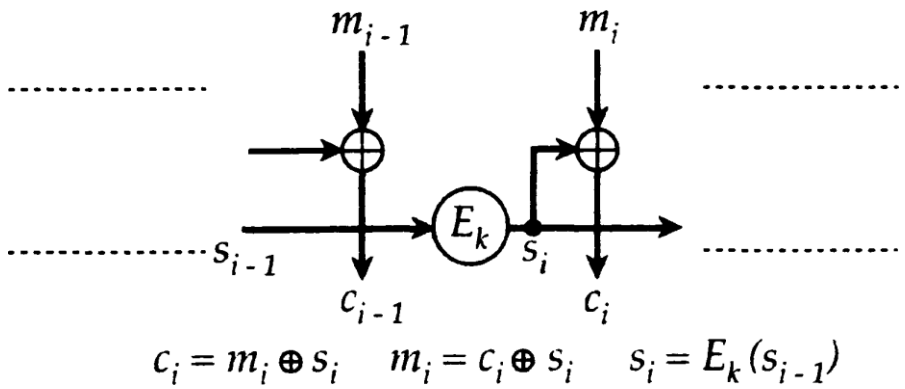


Рисунок 3.5 – Шифрование в режиме OFB

Известные недостатки привели к появлению усовершенствованного варианта шифрования в режиме OFB. Основные изменения касаются метода генерации независимой последовательности блоков: для получения очередного блока предлагается шифровать не s_i, а $s_i + IV \left(\mathrm{mod}\, 2^{64}\right)$, где IV — некоторый вектор инициализации.

Режим шифрования PCBC (Propagating Cipher Block Chaining) применяется в протоколе Kerberos (версия 4) и позволяет обнаруживать ошибки. Данный режим шифрования не является федеральным или международным стандартом. Режим PCBC – вариант режима CBC, обладающий специфическим свойством, в результате дешифрования единичная ошибка распространяется на весь шифротекст (решается обратная задача с точки зрения режима OFB). Данное свойство позволяет с высокой надежностью обнаруживать ошибки, возникающие при передаче сообщений по каналам с шумом. Шифрование в режиме PCBC выполняется по правилу:

$$c_i = E_k\left(m_i \oplus m_{i-1} \oplus c_{i-1}\right),$$

дешифрование:

$$m_i = D_k\left(c_i\right) \oplus c_{i-1} \oplus m_{i-1}$$

где то $m_0 \oplus c_0$ – вектор инициализации.

Стандарты блочного шифрования

Федеральный стандарт США – DES, который ANSI называет Алгоритмом шифрования данных DEA (Data Encryption Algorithm), а ISO — DEA-1, за 20 лет стал мировым стандартом. DES представляет собой блочный шифр, шифрующий данные 64-битовыми блоками. С одного конца алгоритма вводится 64-битовый блок открытого текста, а с другого конца выходит 64-битовый блок шифротекста. DES является симметричным алгоритмом: для шифрования и дешифрования используются одинаковые алгоритм и ключ (за исключением небольших различий в использовании ключа). Длина ключа равна 56 битам. Ключ обычно представляется 64-битовым числом, но каждый восьмой бит используется для проверки четности и игнорируется. Биты четности являются наименьшими значащими битами байтов ключа. Ключ, который может быть любым 56-битовым числом, можно изменить в любой момент времени.

Криптостойкость полностью определяется ключом. Фундаментальным строительным блоком DES является комбинация подстановок и перестановок. DES состоит из 16 циклов (рисунок 6). В общем виде цикл преобразования представлен на рисунке 7.

Если L_i и R_i – левая и правая половины, полученные в результате i-й итерации, K_i – 48-битный ключ для цикла i, а f – функция, выполняющая все подстановки, перестановки и XOR с ключом, то один цикл преобразования можно представить как

$$(L_i, R_i) = (L_{i-1}, R_{i-1} \oplus f(R_{i-1}, K_i)).$$

Учитывая подстановку $F(\square)$ и перестановку $T(\square)$, цикл преобразования можно представить так, как это сделано на рисунке 8. Из рисунка 8 видно, что каждый цикл DES представляет собой композиционный шифр с двумя последовательными преобразованиями – подстановкой $F(\square)$ и перестановкой $T(\square)$ (за исключением последнего, шестнадцатого цикла, где перестановка опускается).

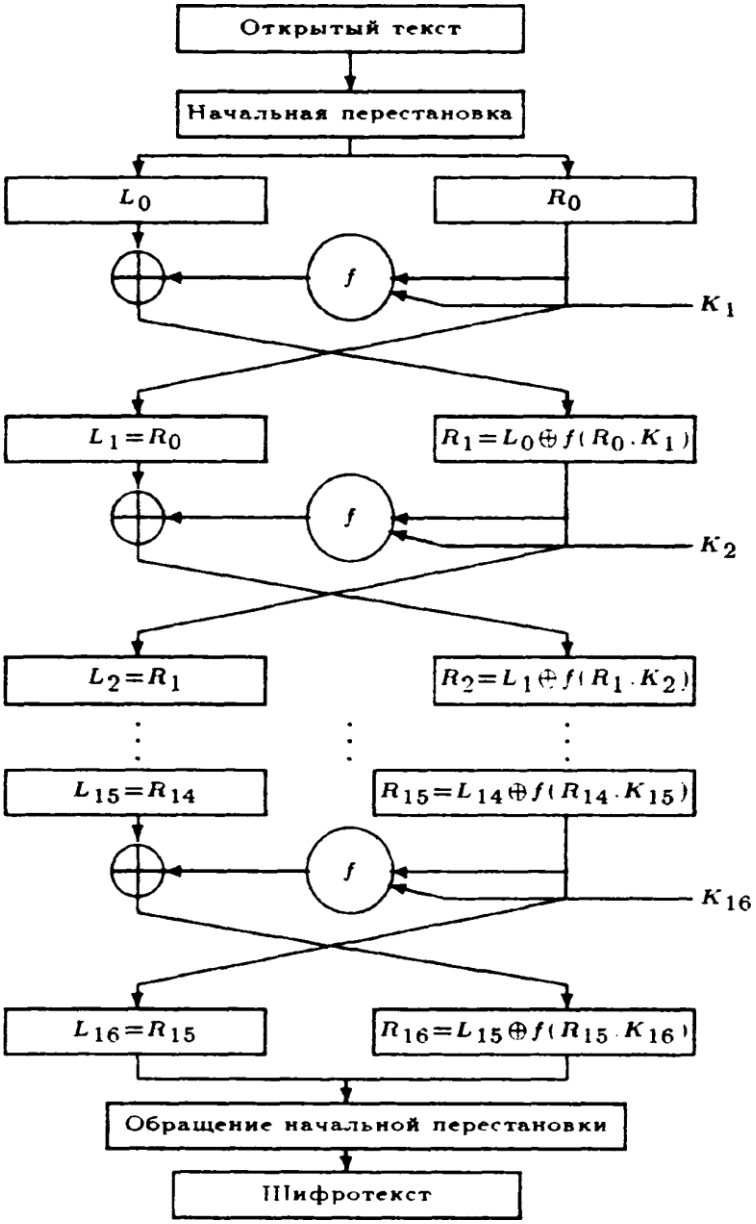


Рисунок 3.6 – Схема DES-преобразования

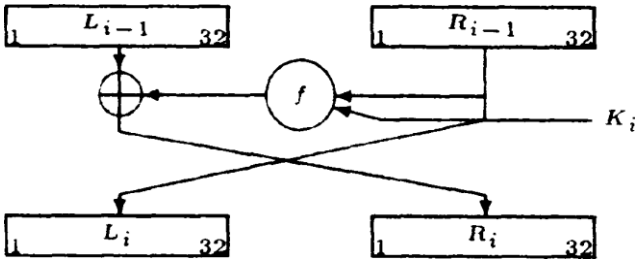


Рисунок 3.7 – Общий цикл DES-преобразования

Таким образом, DES является шифром Фейстеля и сконструирован так, чтобы выполнялось полезное свойство: для шифрования и дешифрования используется один и тот же алгоритм. Единственное отличие состоит в том, что ключи должны использоваться в обратном порядке.

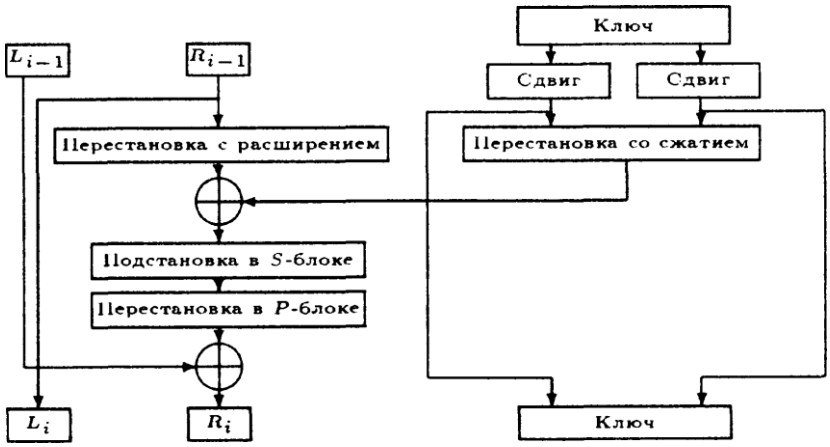


Рисунок 3.8 – Один цикл DES-преобразования

То есть если при шифровании использовались ключи $K_1, K_2, K_3, ..., K_{16}$, то ключами дешифрования будут $K_{16}, K_{15}, K_{14}, ..., K_1$. Алгоритм использует только стандартную арифметику 64-битовых чисел и логические операции, поэтому легко реализуется на аппаратном уровне.

DES работает с 64-битовым блоком открытого текста. После первоначальной перестановки блок разбивается на правую и левую половины длиной по 32 бита. Затем выполняется 16 преобразований (функция *f*), в которых данные объединяются с ключом. После шестнадцатого цикла правая и левая половины объединяются, и алгоритм завершается заключительной перестановкой (обратной по отношению к первоначальной). На каждом цикле (см. рисунок 8) биты ключа сдвигаются, и затем из 56 битов ключа выбираются 48 битов. Правая половина данных увеличивается до 48 битов с помощью перестановки с расширением, объединяется посредством XOR с 48 битами смещенного и переставленного ключа, проходит через 8 S-блоков, образуя 32 новых бита, и переставляется снова. Эти четыре операции и выполняются функцией *f*.

Затем результат функции *f* объединяется с левой половиной с помощью другого XOR. В итоге этих действий появляется новая правая половина, а старая правая становится новой левой половиной. Эти действия повторяются 16 раз, образуя 16 циклов DES.

ГОСТ 28147-89 – это блочный шифр с 256-битным ключом и 32 циклами преобразования, оперирующий 64-битными блоками. В криптоалгоритме также используется дополнительный ключ. Для шифрования открытый текст сначала разбивается на левую и правую половины *L* и *R*. На i-м цикле используется под ключ K_i:

$$L_i = R_{i-1},$$

$$R_i = L_{i-1} \oplus \left(f\left(R_{i-1}, K_i\right)\right).$$

Один цикл криптографического преобразования показан на рисунке 3.9.

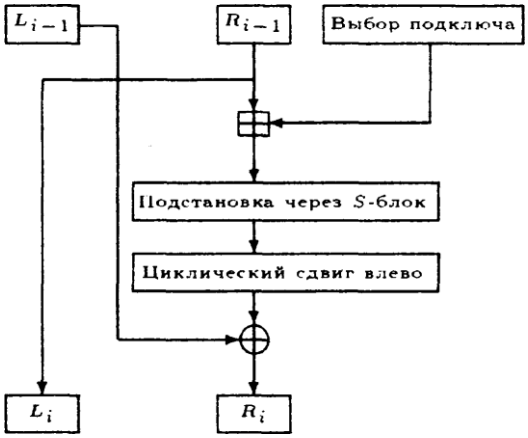


Рисунок 3.9 – Один цикл преобразования ГОСТа 28147-89

Функция f реализована следующим образом. Сначала правая половина и i-й подключ складываются по модулю 2^{32}. Результат разбиваетя на восемь 4-битовых подпоследовательностей, каждая из которых поступает на вход своего S-блока. ГОСТ использует восемь различных S-блоков, первые 4 бита попадают в первый S-блок, вторые 4 бита – во второй S-блок и т.д. Каждый S-блок представляет собой перестановку чисел от 0 до 15. Например, S-блок может выглядеть так: 7,10,2,4,15,9,0,3,6,12,5,13,1,8,11. В этом случае, если на входе S-блока 0, то на выходе 7, если на входе 1, на выходе 10 и т.д. Все восемь S-блоков различны, они фактически являются дополнительным ключевым материалом. Выходы всех восьми S-блоков объединяются в 32-битовое слово, затем все слово циклически сдвигается влево на 11 битов. Наконец, результат объединяется с помощью операции XOR с левой половиной, и получается новая правая половина, а правая половина становится новой левой половиной. Для генерации подключей исходный 256-битный ключ разбивается на восемь 2-битных блоков: $k_1, k_2, k_3, ..., k_8$. На каждом цикле используется свой подключ. Дешифро-

вание выполняется так же, как и шифрование, но инвертируется порядок подключей k_i. Стандарт не определяет способ генерации S-блоков.

Набор S-блоков, указанный в таблице, рекомендуется стандартом ГОСТ Р 34.11-94.

Таблица 3.1 – S-блоки ГОСТа 28147-89

S-блок 1:	4	10	9	2	13	8	0	14	6	11	1	12	7	15	5	3
S-блок 2:	14	11	4	12	6	13	15	10	2	3	8	1	0	7	5	9
S-блок 3:	5	8	1	13	10	3	4	2	14	15	12	7	6	0	9	11
S-блок 4:	7	13	10	1	0	8	9	15	14	4	6	12	11	2	5	3
S-блок 5:	6	12	7	1	5	15	13	8	4	10	9	14	0	3	11	2
S-блок 6:	4	11	10	0	7	2	1	13	3	6	8	5	9	12	15	14
S-блок 7:	13	11	4	1	3	15	5	9	0	10	14	7	6	8	2	12
S-блок 8:	1	15	13	0	5	7	10	4	9	2	3	14	6	11	8	12

Главные различия между DES и ГОСТом заключаются в следующем:

- DES использует сложную процедуру для генерации подключей из ключей. В ГОСТе эта процедура очень проста;
- в DES 56-битный ключ, а в ГОСТе — 256-битный. Если добавить секретные перестановки 5-блоков, то полный объем секретной информации ГОСТа составит примерно 610 бит;
- у S-блоков DES 6-битные входы и 4-битные выходы, а у 5-блоков ГОСТа 4-битные входы и выходы. В обоих алгоритмах используется по восемь S-блоков, но размер S-блока ГОСТа равен четверти размера S-блока DES;
- в DES используются нерегулярные перестановки, названные *P*-блоком, а в ГОСТе используется 11-битный циклический сдвиг влево;

- в DES 16 циклов, а в ГОСТе – 32.

Силовая атака на ГОСТ абсолютно бесперспективна. ГОСТ использует 256-битовый ключ, а если учитывать секретные 5-блоки, то длина ключа будет еще больше. ГОСТ, по-видимому, более устойчив к дифференциальному и линейному криптоанализу, чем DES. Хотя случайные S-блоки ГОСТа при некотором выборе не гарантируют высокой криптостойкости по сравнению с фиксированными 5-блоками DES, их секретность увеличивает устойчивость ГОСТа к дифференциальному и линейному криптоанализу. К тому же эффективность этих криптоаналитических методов зависит от количества циклов преобразования – чем больше циклов, тем труднее криптоанализ. ГОСТ использует в два раза больше циклов, чем DES, что, возможно, приводит к несостоятельности дифференциального и линейного криптоанализа. С точки зрения криптостойкости операция арифметического сложения, используемая в ГОСТе, не хуже, чем операция XOR в DES. Основным различием представляется использование в ГОСТе циклического сдвига вместо перестановки. Перестановка DES увеличивает лавинный эффект. В ГОСТе изменение одного входного бита влияет на один S-блок одного цикла преобразования, который затем влияет на два S-блока □ледующеего цикла, затем на три блока следующего цикла и т.д. Потребуется восемь циклов, прежде чем изменение одного входного бита повлияет на каждый бит результата; в DES для этого нужно только пять циклов. Однако ГОСТ состоит из 32 циклов, а DES только из 16.

Задания на лабораторную работу

Написать программу для реализации блочного шифрования файла:

№ вв	Алгоритм блочного шифрования
1,2,3,4	ГОСТ 28147-89
5,6,7,8	BlowFish
9,10,11,12	3-Way
13,14,15	RC5

Контрольные вопросы

1. Что такое симметричное шифрование?

2. В чем особенность блочных шифров?

3. Какова длина ключа блочного шифра?

4. На чем базируется криптостойкость блочного шифра?

5. Какие элементарные операции используются в симметричном шифровании?

Лабораторная работа №4

Разработка программного средства для реализации алгоритма асимметричного шифрования и дешифрования

Цель работы: изучить принципы построения асимметричных криптосистем, получить навыки разработки программ для реализации асимметричного шифрования и дешифрования.

Программа работы

1. Изучить принципы асимметричного шифрования и стандарты асимметричного шифрования.

2. Программная реализация асимметричного шифрования и дешифрования.

3. Составить и защитить отчет по результатам работы.

Краткие сведения из теории

Криптосистема RSA, предложенная в 1977 г. Ривестом (R. Rivest), Шамиром (A. Shamir) и Адлеманом (L. Adleman), предназначена для шифрования и цифровой подписи. Для генерации парных ключей используются два больших случайных простых числа, p и q. В целях максимальной криптостойкости p и q выбираются равной длины. Затем вычисляется произведение: $n = pq$.

Далее случайным образом выбирается ключ шифрования e, такой, что e и $\phi(n) = (p-1)(q-1)$ являются взаимно простыми числами. Наконец расширенный алгоритм Евклида используется для вычисления ключа дешифрования d, такого, что $ed \equiv 1 \bmod \phi(n)$. Другими словами,

$$d = e^{-1} \bmod \phi(n).$$

Заметим, что d и n – также взаимно простые числа. Числа e и n – открытый ключ, а d – секретный. Два простых числа p и q хранятся в секрете. Для шифрования сообщения m необходимо выполнить его разбивку на блоки, каждый из которых меньше n (для двоичных данных выбирается самая большая степень числа 2, меньшая n). То есть если p и q – 100-разрядные простые числа, то n будет содержать около 200 разрядов и каждый блок сообщения m_i должен иметь такое же число разрядов. Если нужно зашифровать фиксированное число блоков, их можно дополнить несколькими нулями слева, чтобы гарантировать, что блоки всегда будут меньше n. Зашифрованное сообщение c будет состоять из блоков a той же самой длины. Шифрование сводится к вычислению $c_i = m_i^e \bmod n$. При дешифровании для каждого зашифрованного блока c_i вычисляется $m_i = c_i^d \bmod n$. Все вычисления выполняются

по mod *n*. Сообщение может быть зашифровано с помощью *d*, а дешифровано с помощью e, возможен любой выбор.

Численный пример

$p = 47$ и $q = 71$, то $n = pq = 3337$. Ключ e не должен иметь общих множителей с $\phi(n) = 46 \times 72 = 3220$. Выбираем (случайно) e равным 79. Тогда $d = 79^{-1} \bmod 3220 = 1019$. Публикуем e и n, сохранив в секрете d. Для шифрования сообщения $m = 6882326879666683$ сначала разделим его на блоки. Для выбранных параметров ограничимся блоками по три десятичных разряда. Сообщение разбивается на шесть блоков m_i: $m_1 = 688$, $m_2 = 232$, $m_3 = 687$, $m_4 = 966$, $m_5 = 668$, $m_6 = 003$. Первый блок шифруется как $688^{79} \bmod 3337 = 1570 = c_1$. Выполняя те же операции для последующих блоков, создадим шифротекст сообщения:

$$c = 1570275620912276242 3158.$$

Для дешифрования нужно выполнить возведение в степень, используя ключ дешифрования 1019:

$$1570^{1019} \bmod 3337 = 688 = m_1.$$

Аналогично восстанавливается оставшаяся часть сообщения.

Предполагается, что криптостойкость RSA зависит от проблемы разложения на множители больших чисел. Однако никогда не было доказано математически, что нужно разложить n на множители, чтобы восстановить m по c и e. Не исключено, что может быть открыт совсем иной способ криптоанализа RSA. Однако, если этот новый способ позволит криптоаналитику получить d, он также может быть использован для разложения на множители больших чисел. Также можно атаковать RSA, угадав значение (p – l)(q – 1). Однако этот метод не проще разложения n на множители. При использовании RSA раскрытие даже не-

скольких битов информации по шифротексту не легче, чем дешифрование всего сообщения. Самой очевидной атакой на RSA является разложение n на множители. Любой противник сможет получить открытый ключ e и модуль n. Чтобы найти ключ дешифрования d, противник должен разложить n на множители. Криптоаналитик может перебирать все возможные d, пока не подберет правильное значение. Но подобная силовая атака даже менее эффективна, чем попытка разложения n на множители.

Некоторые атаки используют уязвимость криптографического протокола. Важно понимать, что само по себе использование RSA не обеспечивает требуемого уровня безопасности системы. Рассмотрим несколько сценариев.

Сценарий 1

Злоумышленнику удалось перехватить сообщение c, зашифрованное с помощью открытого RSA-ключа абонента А. Он хочет прочитать сообщение. Для раскрытия $m = c^d$ он сначала выбирает первое случайное число r, меньшее n, и затем, воспользовавшись открытым ключом Алисы e, вычисляет $x = r^e \bmod n$, $y = xc \bmod n$, $t = r^{-1} \bmod n$. Если $x = r^e \bmod n$, то $r = x^d \bmod n$.

Далее злоумышленник вынуждает абонента А подписать сообщение y. Таким образом, процедура вычисления подписи на секретном ключе соответствует процедуре дешифрования сообщения y. Абонент должен подписать сообщение, а не значение хэш-функции. Такой обман вполне реален, так как абонент А никогда раньше не видел y. Абонент А посылает злоумышленнику $u = y^d \bmod n$. Теперь злоумышленник раскрывает m, вычисляя

$$tu \bmod n = r^{-1} y^d \bmod n = r^{-1} x^d c^d \bmod n = m.$$

Сценарий 2

Если абонент А хочет заверить документ, он посылает его нотариусу. Нотариус подписывает его цифровой подписью и отправляет обратно. При этом хэш-функции не используются, нотариус шифрует все сообщение на своем секретном ключе. Злоумышленник хочет, чтобы нотариус подписал такое сообщение, которое в обычном случае тот никогда не подпишет. Это может быть фальшивая временная метка, либо автором этого сообщения может являться другое лицо. Какой бы ни была причина, нотариус никогда не подпишет это сообщение, если у него будет возможность выбора. Назовем это сообщение m'. Сначала злоумышленник выбирает произвольное значение x и вычисляет $y = x^e \bmod n$. Параметр e он может получить без труда – это открытый ключ нотариуса, и должен быть опубликован для проверки подписи последнего. Теперь злоумышленник вычисляет $m = ym' \bmod n$ и посылает m нотариусу на подпись. Нотариус возвращает $m^d \bmod n$. Далее злоумышленник вычисляет $\left(m^d \bmod n\right) x^{-1} \bmod n$, которое равно $m'^d \bmod n$ и является подписью m'.

Сценарий 3

Злоумышленник хочет, чтобы абонент подписал некое сообщение m_3. Для этого он создает два сообщения, m_1 и m_2, такие, что $m_3 = m_1 m_2 \bmod n$. Если злоумышленник заставит абонента подписать m_1 и m_2, то сможет вычислить подпись для m_3:

$$m_3^d = \left(m_1^d \bmod n\right)\left(m_2^d \bmod n\right) \bmod n.$$

Вывод – никогда нельзя использовать RSA для подписи случайных документов. Применение хэш-функций в технологии RSA-подписи строго обязательно.

Сценарий 4

При реализации RSA можно попробовать раздать всем абонентам криптосети одинаковый модуль n, но каждому – свои значения показателей степени e и d . При этом наиболее очевидная проблема заключается в том, что если одно и то же сообщение когда-нибудь шифровалось разными показателями степени (при фиксированном модуле) и эти два показателя – взаимно-простые числа (как обычно и бывает), то открытый текст может быть раскрыт даже при неизвестных ключах дешифрования. Пусть заданы: m – открытый текст, e_1 и e_2 – два ключа шифрования, n – общий модуль. Шифротекстами сообщения являются:

$$c_1 = m^{e_1} \bmod n,\ c_2 = m^{e_2} \bmod n.$$

Криптоаналитик знает n, e_1, e_2, c_1 и c_2. Так как e_1 и e_2 – взаимно-простые числа, то, воспользовавшись расширенным алгоритмом Евклида, можно найти такие числа r и s, что

$$re_1 + se_2 = 1.$$

Полагая r отрицательным (или r, или s должно быть отрицательным), можно снова воспользоваться расширенным алгоритмом Евклида для вычисления c_1^{-1}. Тогда

$$\left(c_1^{-1}\right)^{-r} c_2^s = m \bmod n.$$

Сценарий 5

Известно, что криптосистема RSA обладает низкой криптостойкостью при зашифрованном на малом e коротком сообщении. Действительно, при $c = m^e < n$ открытый текст m может быть восстановлен по шифротексту c при помощи процедуры извлечения корня. Фактически подобная атака возможна и тогда, когда в процессе возведения в степень

выполнялось некоторое количество приведений по модулю. При $c > n$ трудоемкость такой атаки ниже трудоемкости исчерпывающего перебора для m. Однако меры противодействия также очевидны, – либо открытый ключ e должен быть достаточно большим, либо открытый текст не должен быть коротким. Выбор малого e обусловлен соображениями вычислительной эффективности шифрования и проверки подписи. Таким образом, разумный подход заключается в искусственном наращивании коротких открытых текстов («набивки»). При этом необходимо следить за тем, чтобы удлиненный открытый текст при числовом отображении не превращался в набор множителей некоторого известного числа P, например $P = 2^l$, что происходит при дополнении открытого текста последовательностью нулей справа (со стороны младших разрядов).

На основании перечисленных атак можно сформулировать следующие ограничения при использовании RSA :

- знание одной пары показателей шифрования/дешифрования для данного модуля позволяет злоумышленнику разложить модуль на множители;

- знание одной пары показателей шифрования/дешифрования для данного модуля позволяет злоумышленнику вычислить другие пары показателей, не раскладывая модуль на множители;

- в криптографических протоколах с использованием RSA общий модуль использоваться не должен;

- для предотвращения раскрытия малого показателя шифрования сообщения должны быть дополнены («набиты») случайными значениями;

- показатель дешифрования должен быть большим.

Недостаточно использовать криптостойкий алгоритм, безопасной должна быть вся криптосистема, включая криптографический протокол. Слабое место любого из трех этих компонентов сделает небезопасной всю систему.

Задания на лабораторную работу

№ варианта	Задание № 1 Написать программу для реализации RSA шифрования/ дешифрования	Задание № 2 Произвести моделирование атаки на RSA криптосистему
1	607 613	Сценарий 1
2	617 619	Сценарий 2
3	631 641	Сценарий 3
4	643 647	Сценарий 4
5	653 659	Сценарий 5
6	661 673	Сценарий 1
7	677 683	Сценарий 2
8	691 701	Сценарий 3
9	709 719	Сценарий 4
10	727 733	Сценарий 5
11	739 743	Сценарий 1
12	751 757	Сценарий 2
13	761 769	Сценарий 3
14	773 787	Сценарий 4
15	797 809	Сценарий 5

Контрольные вопросы

1. В чем особенность асимметричных систем шифрования?
2. На чем базируется криптостойкость RSA?
3. Как увеличить производительность системы шифрования RSA?
4. Какие атаки на систему RSA вам известны?
5. Как противодействовать атакам на систему RSA?

Лабораторная работа №5

Криптосистема ЭльГамаля

Цель работы: изучить принципы построения асимметричных криптосистем, получить навыки разработки программ для реализации асимметричного шифрования и дешифрования.

Программа работы:

1. Изучить принципы асимметричного шифрования и стандарты асимметричного шифрования.

2. Программная реализация асимметричного шифрования и дешифрования.

3. Составить и защитить отчет по результатам работы.

Краткие сведения из теории

Криптосистему, предложенную ЭльГамалем (T. ElGamal) в 1985 г., можно использовать как для цифровых подписей, так и для шифрования. Криптостойкость определяется трудоемкостью вычисления дискретного логарифма над конечным полем.

Для генерации пары ключей сначала выбираются простое число p и два случайных числа, g и x; оба этих числа должны быть меньше p. Затем вычисляется

$$y = g^x \bmod p.$$

Открытым ключом являются y, g и p. И g, и p можно сделать общими для группы пользователей. Секретным ключом является x.

Вычисление и проверка подписи.

Чтобы подписать сообщение M, сначала выбирается случайное число k, взаимно простое с $(p-1)$. Затем вычисляется $a = g^k \bmod p$, и с помощью расширенного алгоритма Евклида из уравнения

$M = (xa + kb) \bmod (p-1)$ находится b. Подписью является пара чисел: a и b. Случайное значение k должно храниться в секрете. Для проверки подписи необходимо убедиться, что

$$y^a x^b \bmod p = g^M \bmod p.$$

Каждая новая подпись требует нового значения k, и это значение должно выбираться случайным образом. Если злоумышленник раскроет k, используемое абонентом, он сможет раскрыть секретный ключ x. Если злоумышленник сможет получить два сообщения, подписанные при помощи одного и того же k, он сможет раскрыть x, даже не зная k.

Численный пример

Выберем $p = 11$ и $g = 2$. Пусть секретный ключ $x = 8$. Вычислим

$$y = g^x \bmod p = 256 \bmod 11 = 3.$$

Открытым ключом являются $y = 3$, $g = 2$ и $p = 11$. Чтобы подписать $M = 5$, сначала выберем случайное число $k = 9$. Убедимся, что $\gcd(9,10) = 1$. Далее вычислим $a = g^k \bmod p = 512 \bmod 11 = 6$. Затем с помощью расширенного алгоритма Евклида найдем b из уравнения $M = (xa + kb) \bmod (p-1)$:

$$5 = (8 \cdot 6 + 9 \cdot b) \bmod 10.$$

Решение: $b = 3$, а подпись представляет собой пару: $a = 6$ и $b = 3$. Для проверки подписи убедимся, что $y^a x^b \bmod p = g^M \bmod p$:

$$3^6 6^3 \bmod 11 = 32 \bmod 11.$$

Шифрование/дешифрование.

Для шифрования сообщения M сначала выбирается случайное число k, взаимно-простое с $(p-1)$. Затем вычисляются $a = g^k \bmod p$, $b = y^k M \bmod p$.

Пара (a, *b)* является шифротекстом.

Для дешифрования (a, b) вычисляется

$$M = \frac{b}{a^x} \bmod p.$$

Преобразование обратимо, так как $a^x \equiv g^{kx} \bmod p$.

По сути описанное преобразование – это то же самое, что и экспоненциальный ключевой обмен по Диффи-Хеллману, за исключением того, что y – это часть ключа, а при шифровании сообщение умножается на y^k.

Метод экспоненциального ключевого обмена Диффи-Хеллмана

Метод экспоненциального ключевого обмена Диффи-Хеллмана – первая криптосистема с открытым ключом – был изобретен Диффи (W. Diffie) и Хеллманом (M. Hellman) в 1976 году. Криптостойкость метода определяется трудоемкостью вычисления дискретного логарифма. Метод может быть использован для распределения ключей – два абонента могут воспользоваться им для генерации общего секретного ключа, но его нельзя использовать для шифрования и дешифрования сообщений.

Сначала абоненты А и Б вместе выбирают большие простые числа n и g, так, чтобы g было примитивным элементом в конечном поле *GF(n)*. Эти два целых числа хранить в секрете необязательно, абоненты могут договориться об их использовании по несекретному каналу. Эти числа могут даже совместно использоваться группой пользователей. Затем реализуется следующий протокол:

- абонент А выбирает случайное большое целое число x и посылает абоненту Б

$$X = g^x \bmod n;$$

- абонент Б выбирает случайное большое целое число y и посылает абоненту А

$$Y = g^y \bmod n;$$

- абонент А вычисляет

$$k = Y^x \bmod n;$$

- абонент Б вычисляет

$$k' = X^y \bmod n.$$

И k, и k' равны $g^{xy} \bmod n$. Никто из прослушивающих этот канал злоумышленников не сможет вычислить это значение, им известны только n, g, X и Y, и не известны x и y. Для получения x и y необходимо вычислить дискретный логарифм. Таким образом, k – это секретный ключ, который абоненты вычисляют независимо. Выбор g и n может заметно влиять на криптостойкость. n должно быть обязательно большим числом: криптостойкость зависит также от трудоемкости разложения на множители чисел того же размера, что и n. Можно выбирать любое g, являющееся примитивным элементом; нет причин, по которым нельзя было бы выбрать наименьшее возможное g. На самом деле число g может даже и не быть примитивным элементом, оно лишь должно порождать достаточно большую подгруппу мультипликативной группы в $GF(n)$.

Без дополнительных мер безопасности (введения сертификатов открытых ключей) описанный метод ключевого обмена уязвим с точки зрения атаки, известной под названием «человек посередине» (man-in-the-middle attack).

Предположим, злоумышленник может не только подслушивать сообщения абонентов, но и изменять и удалять сообщения, а также создавать совершенно новые ложные сообщения. Тогда злоумышленник может выдавать себя за абонента Б, сообщающего что-то абоненту А, или за А, сообщающего что-то Б. Атака состоит из следующих действий:

1. абонент А посылает абоненту Б свой открытый ключ. Злоумышленник перехватывает его и посылает Б свой собственный открытый ключ;

2. Б посылает А свой открытый ключ. Злоумышленник перехватывает его и посылает А свой собственный открытый ключ;

3. когда абонент А посылает сообщение абоненту Б, зашифрованное на его открытом ключе, злоумышленник его перехватывает. Так как сообщение в действительности зашифровано на открытом ключе злоумышленника, он расшифровывает его, снова зашифровывает на открытом ключе абонента Б и посылает его Б;

4. когда Б посылает сообщение абоненту А, зашифрованное на ее открытом ключе, злоумышленник его перехватывает. Так как сообщение в действительности зашифровано на открытом ключе злоумышленника, он расшифровывает его и затем снова зашифровывает на открытом ключе А и посылает абоненту А.

Атака возможна, даже если открытые ключи А и Б хранятся в базе данных. Злоумышленник может перехватить запрос А к базе данных и подменить открытый ключ Б своим собственным. То же самое он может сделать и с открытым ключом абонента А. Злоумышленник может атаковать базу данных и подменить открытые ключи Б и А своими собственными. Затем, дождавшись, когда абоненты начнут обмениваться сообщениями, он выполняет перехват и подмену. Подобная атака весьма эффективна, так как у А и Б нет возможности проверить, действительно ли они общаются именно друг с другом. Если вмешательство злоумышленника не приводит к заметным задержкам при передаче сообщений, абоненты не смогут обнаружить, что кто-то, расположенный между ними, читает все их секретные сообщения.

Каждый абонент криптосети может опубликовать свой открытый ключ $X = g^x \bmod n$ в общей базе данных. Если абонент А захочет установить связь с абонентом Б, ему понадобится только получить открытый ключ Б и затем сгенерировать общий секретный ключ. Он может зашифровать сообщение этим ключом и послать его Б. Абонент Б извлечет открытый ключ А и вычислит общий секретный ключ. Каждая пара абонентов может использовать уникальный секретный ключ; не требуется никаких предварительных обменов данными между ними. Открытые ключи должны пройти сертификацию, чтобы предотвратить атаки, связанные с подменой ключей (в первую очередь для противодействия атаке «человек посередине»), и должны регулярно меняться.

Протокол ключевого обмена для нескольких участников

Описанный выше протокол ключевого обмена легко можно расширить для случая трех и более участников. В приводимом ниже примере А, Б и В вместе генерируют общий секретный ключ.

А выбирает случайное большое целое число x и вычисляет

$$X = g^x \bmod n.$$

Б выбирает случайное большое целое число y и посылает абоненту В

$$Y = g^y \bmod n.$$

К выбирает случайное большое целое число z и посылает абоненту А

$$Z = g^z \bmod n.$$

А посылает Б

$$Z' = Z^x \bmod n.$$

Б посылает В

$$X' = X^y \bmod n.$$

В посылает А

$$Y' = Y^z \bmod n.$$

А вычисляет

$$k = Y'^x \bmod n.$$

Б вычисляет

$$k = Z'^y \bmod n.$$

В вычисляет

$$k = X'^z \bmod n.$$

Секретный ключ k равен $g^{xyz} \bmod n$, и никто из подслушивающих каналы злоумышленников не сможет вычислить это значение. Протокол можно легко расширить для четырех и более участников, просто добавляются участники и этапы вычислений.

Односторонняя генерация ключа

Этот вариант метода Диффи-Хеллмана позволяет А сгенерировать ключ и послать его Б.

А выбирает случайное большое целое число x и генерирует

$$k = g^x \bmod n.$$

Б выбирает случайное большое целое число y и посылает А

$$Y = g^y \bmod n.$$

А посылает Б

$$X = Y^x \bmod n.$$

Б вычисляет

$$z = y^{-1}$$

$$k' = X^z \bmod n.$$

Если все выполнено правильно, то $k = k'$. Преимуществом этого метода состоит в том, что k можно вычислить заранее, до взаимодействия, и абонент А может шифровать сообщения с помощью k задолго до уста-

новления соединения с Б. Он может послать сообщение сразу множеству абонентов, а передать ключ позднее – каждому в отдельности.

Задание на лабораторную работу

1. Разработать программу для шифрования/дешифрования по схеме ЭльГамаля.

№ варианта	Модуль криптографической схемы
1	1009
2	1013
3	1019
4	1021
5	1031
6	1033
7	1039
8	1049
9	1051
10	1061
11	1063
12	1069
13	1087
14	1091
15	1093

2. Разработать две программы, которые, устанавливая связь с применением сокетов, реализуют одностороннюю генерацию ключа.

Контрольные вопросы

1. Назначение цифровой подписи.

2. В чем отличие криптосхемы ЭльГамаля от RSA?

3. На чем базируется криптостойкость системы ЭльГамаля?

Литература

1 Скрипник Д.А. Общие вопросы технической защиты информации. — Электрон. текст. дан. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — Режим доступа : http ://www.iprbookshop.ru/52161.— ЭБС «IPRbooks», по паролю.

2 Лапонина О.Р. Основы сетевой безопасности. Криптографические алгоритмы и протоколы взаимодействия. — Электрон. текст. дан.— М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — Режим доступа : http ://www.iprbookshop.ru/52217.— ЭБС «IPRbooks», по паролю.

3 Алексеев, А.П.; Многоуровневая защита информации Электронный ресурс : монография / А.П. Алексеев. - Самара : Поволжский государственный университет телекоммуникаций и информатики, 2017. - 128 с. - Книга находится в базовой версии ЭБС IPRbooks. - ISBN 978-5-904029-72-2

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
НЕВИННОМЫССКИЙ ТЕХНОЛОГИЧЕСКИЙ ИНСТИТУТ (ФИЛИАЛ)

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ПРАКТИЧЕСКИМ ЗАНЯТИЯМ

по дисциплине

"Информационная безопасность и защита данных"

Направление подготовки 15.04.04

«Автоматизация технологических процессов и производств»

Направленность (профиль) «Информационно-управляющие системы»

Форма обучения - очно-заочная

Год начала обучения 2022

Реализуется в 5 семестре

Невинномысск 2022

Содержание:

Цель и задачи освоения дисциплины (модуля)

Целью является формирование набора профессиональных компетенций будущего магистра по направлению подготовки 15.04.04 Автоматизация технологических процессов и производств.

Задачи изучения дисциплины заключаются в приобретении студентами знаний и практических навыков в области, определяемой основной целью дисциплины.

Наименование компетенций

Код	Формулировка
ПК-9	Способность обеспечивать надежность и безопасность на всех этапах жизненного цикла продукции, выбирать системы экологической безопасности производства

Практическая работа №1
по учебной дисциплине **«Защита информации в системах управления»**

Раздел 1. Информационная безопасность информационных систем

Идентификация, аутентификация и управление доступом Криптографическая защита информации.

1.Цель работы

1. Изучить способы исследования методов аутентификации
2. Исследовать способы управления доступом

2. Подготовка к занятию

1. Изучить (повторить) теоретический материал.
2. Ознакомиться с программой лабораторной работы.
3. Подготовить отчет о лабораторной работе.
4. Ответить на контрольные вопросы.

3. Распределение времени занятия:

Всего: 90 мин
Вступительная часть 2 мин
Проверка готовности студентов к занятию 5 мин
Основная часть – 70 мин
Проверка выполнения практического занятия 10 мин
Заключительная часть 3 мин

4. Правила работы в лаборатории

К работе в лаборатории допускаются лица, изучившие правила и меры безопасности, сдавшие зачет по ним и усвоившие порядок выполнения лабораторной работы.

4.1. Требования безопасности перед началом работ

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории.

- Убедиться в целостности электрических розеток и разъемов. В лаборатории необходимо быть в сменной обуви.

- Включение компьютера производить только после получения допуска

по выполняемой работе и разрешения преподавателя или лаборанта.

4.2. Требования безопасности во время работы

- выполняя практическое занятие, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и носители информации, непосредственно относящиеся к данному лабораторному занятию;
- подключение и отключение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса;
- при обнаружении неисправностей в оборудовании немедленно отключить источники питания и доложить об этом руководителю занятий или лаборанту.

4.3. Требования безопасности по окончании работы

- доложить руководителю занятий или лаборанту о завершении работ;
- привести в порядок и сдать рабочее место лаборанту, и доложить руководству

Парольная защита

Под **несанкционированным доступом к информации** (НСД) согласно руководящим документам Гостехкомиссии будем понимать доступ к информации, нарушающий установленные правила разграничения доступа и осуществляемый с использованием штатных средств, предоставляемых СВТ или АС. НСД может носить случайный или намеренный характер.

Можно выделить несколько обобщенных категорий методов защиты от НСД, в частности:

1. организационные;
2. технологические;
3. правовые.

К первой категории относятся меры и мероприятия, регламентируемые внутренними инструкциями организации, эксплуатирующей информационную систему. Пример такой защиты — присвоение грифов секретности документам и материалам, хранящимся в отдельном помещении, и контроль доступа к ним сотрудников. Вторую категорию составляют механизмы защиты, реализуемые на базе программно-аппаратных средств, например систем идентификации и

аутентификации или охранной сигнализации. Последняя категория включает меры контроля за исполнением нормативных актов общегосударственного значения, механизмы разработки и совершенствования нормативной базы, регулирующей вопросы защиты информации. Реализуемые на практике методы, как правило, сочетают в себе элементы нескольких из перечисленных категорий. Так, управление доступом в помещения может представлять собой взаимосвязь организационных (выдача допусков и ключей) и технологических (установку замков и систем сигнализации) способов защиты.

Рассмотрим подробнее такие взаимосвязанные методы защиты от НСД, как идентификация, аутентификация и используемое при их реализации криптографическое преобразование информации.

Идентификация — это присвоение пользователям идентификаторов и проверка предъявляемых идентификаторов по списку присвоенных.

Аутентификация — это проверка принадлежности пользователю предъявленного им идентификатора. Часто аутентификацию также называют подтверждением или проверкой подлинности.

Под безопасностью (стойкостью) системы идентификации и аутентификации будем понимать степень обеспечиваемых ею гарантий того, что злоумышленник не способен пройти аутентификацию от имени другого пользователя, В этом смысле, чем выше стойкость системы аутентификации, тем сложнее злоумышленнику решить указанную задачу. Система идентификации и аутентификации является одним из ключевых элементов инфраструктуры защиты от НСД любой информационной системы.

Различают три группы методов аутентификации, основанных на наличии у каждого пользователя:

1. индивидуального объекта заданного типа;
2. знаний некоторой известной только ему и проверяющей стороне информации;
3. индивидуальных биометрических характеристик.

К первой группе относятся методы аутентификации, использующие

удостоверения, пропуска, магнитные карты и другие носимые устройства, которые широко применяются для контроля доступа в помещения, а также входят в состав программно-аппаратных комплексов защиты от НСД к средствам вычислительной техники.

Во вторую группу входят методы аутентификации, использующие пароли. По экономическим причинам они включаются в качестве базовых средств защиты во многие программно-аппаратные комплексы защиты информации. Все современные операционные системы и многие приложения имеют встроенные механизмы парольной защиты.

Последнюю группу составляют методы аутентификации, основанные на применении оборудования для измерения и сравнения с эталоном заданных индивидуальных характеристик пользователя: тембра голоса, отпечатков пальцев, структуры радужной оболочки глаза и др. Такие средства позволяют с высокой точностью аутентифицировать обладателя конкретного биометрического признака, причем "подделать" биометрические параметры практически невозможно. Однако широкое распространение подобных технологий сдерживается высокой стоимостью необходимого

оборудования.

Если в процедуре аутентификации участвуют только две стороны, устанавливающие подлинность друг друга, такая процедура называется непосредственной аутентификацией (си rect passnrOrd authenti Ceti ОН). Если же в процессе аутентификации участвуют не только эти стороны, но и другие, вспомогательные, говорят об аутентификации с участием доверенной стороны (trusted thi гси рагу authenti сан оп). При этом третью сторону называют сервером аутентификации (authenti Ceti оп server) или арбитром (аГbitretor).

Наиболее распространенные методы аутентификации основаны на применении многоразовых или одноразовых паролей. Из-за своего широкого распространения и простоты реализации парольные схемы часто в первую очередь становятся мишенью атак злоумышленников. Эти методы включают следующие

разновидности способов аутентификации:

— по хранимой копии пароля или его свёртке (p1 ai ntext-equi a] еп);
— по некоторому проверочному значению (veri f i er-based) ;
— без непосредственной передачи информации о пароле проверяющей стороне (zero-
knowl edge) ;
— с использованием пароля для получения криптографического ключа (CryptographiC).

В первую разновидность способов входят системы аутентификации, предполагающие наличие у обеих сторон копии пароля или его свертки. Для организации таких систем требуется создать и поддерживать базу данных, содержащую пароли или сверки паролей всех пользователей. Их слабой стороной является то, что получение злоумышленником этой базы данных позволяет ему проходить аутентификацию от имени любого пользователя.

Способы, составляющие вторую разновидность, обеспечивают более высокую степень безопасности парольной системы, так как проверочные значения, хотя они и зависят от паролей, не могут быть непосредственно использованы злоумышленником для аутентификации.

Наконец, аутентификация без предоставления проверяющей стороне какой бы то ни было информации о пароле обеспечивает наибольшую степень защиты. Этот способ гарантирует безопасность даже в том случае, если нарушена работа проверяющей стороны (например, в программу регистрации в системе внедрен "троянский конь").

Особым подходом в технологии проверки подлинности являются криптографические протоколы аутентификации. Такие протоколы описывают последовательность действий, которую должны совершить стороны для взаимной аутентификации, кроме того, эти действия, как правило, сочетаются с генерацией и распределением криптографических ключей для шифрования последующего информационного обмена. Корректность протоколов аутентификации вытекает из свойств

задействованных в них математических и криптографических преобразований и может быть строго доказана. Обычные парольные системы проще и дешевле для реализации, но менее безопасны, чем системы с криптографическими протоколами. Последние обеспечивают более надежную защиту и дополнительно решают задачу распределения ключей. Однако используемые в них технологии могут быть объектом законодательных ограничений.

Для более детального рассмотрения принципов построения парольных систем сформулируем несколько основных определений.

Идентификатор пользователя - некоторое уникальное количество информации, позволяющее различать индивидуальных пользователей парольной системы (проводить их идентификацию). Часто идентификатор также называют именем пользователя или именем учетной записи пользователя.

Пароль пользователя - некоторое секретное количество информации, известное только пользователю и парольной системе, которое может быть запомнено пользователем и предъявлено для прохождения процедуры аутентификации. Одноразовый пароль дает возможность пользователю однократно пройти аутентификацию. Многоразовый пароль может быть использован для проверки подлинности повторно.

Учетная запись пользователя - совокупность его идентификатора и его пароля. База данных пользователей парольной системы содержит учетные записи всех пользователей данной парольной системы.

Под парольной системой будем понимать программно-аппаратный комплекс, реализующий системы идентификации и аутентификации пользователей АС на основе одноразовых или многоразовых паролей. Как правило, такой комплекс функционирует совместно с подсистемами разграничения доступа и регистрации событий. В отдельных случаях парольная система может выполнять ряд дополнительных функций, в частности генерацию и распределение кратковременных (сеансовых)

криптографических ключей.

Основными компонентами парольной системы являются:

— интерфейс пользователя;

— интерфейс администратора;

— модуль сопряжения с другими подсистемами безопасности;

— база данных учетных записей.

Парольная система представляет собой "передний край обороны" всей системы безопасности. Некоторые ее элементы (в частности, реализующие интерфейс пользователя) могут быть расположены в местах, открытых для доступа потенциальному злоумышленнику. Поэтому парольная система становится одним из первых объектов атаки при вторжении злоумышленника в защищенную систему. Ниже перечислены типы угроз безопасности парольных систем:

1. Разглашение параметров учетной записи через:

— подбор в интерактивном режиме;

— подсматривание;

— преднамеренную передачу пароля его владельцем другому лицу;

— захват базы данных парольной системы (если пароли не хранятся в базе в открытом виде, для их восстановления может потребоваться подбор или дешифрование);

— перехват переданной по сети информации о пароле;

— хранение пароля в доступном месте.

2. Вмешательство в функционирование компонентов парольной системы через:

— внедрение программных закладок;

— обнаружение и использование ошибок, допущенных на стадии разработки;

— выведение из строя парольной системы.

Некоторые из перечисленных типов угроз связаны с наличием так называемого человеческого фактора, проявляющегося в том, что

пользователь может:

— выбрать пароль, который легко запомнить и также легко подобрать;

— записать пароль, который сложно запомнить, и положить запись в доступном месте;

— ввести пароль так, что его смогут увидеть посторонние:

— передать пароль другому лицу намеренно или под влиянием заблуждения.

В дополнение к выше сказанному необходимо отметить существование "парадокса человеческого фактора". Заключается он в том, что пользователь нередко стремится выступать скорее противником парольной системы, как, впрочем, и любой системы безопасности, функционирование которой влияет на его рабочие условия, нежели союзником системы защиты, тем самым ослабляя ее. Защита от указанных угроз основывается на ряде перечисленных ниже организационно-технических мер и мероприятий.

Выбор паролей

В большинстве систем пользователи имеют возможность самостоятельно выбирать пароли или получают их от системных администраторов. При этом для уменьшения деструктивного влияния описанного выше человеческого фактора необходимо реализовать ряд требований к выбору и использованию паролей.

Таблица 1

Требование к выбору пароля	Получаемый эффект
Установление минимальной длины пароля	Усложняет задачу злоумышленника при попытке подсмотреть пароль или подобрать пароль методом <сготального опробования»
Использование в пароле различных групп символов	Усложняет задачу злоумышленника при попытке подобрать пароль методом «тотального опробования»
Проверка и отбраковка пароля по словарю	Усложняет задачу злоумышленника при попытке подобрать пароль по словарю

Установление максимального срока действия пароля	Усложняет задачу злоумышленника при попытке подобрать пароль методом «тотального опробования», в том числе без непосредственного обращения к системе защиты (режим Off-11 Пе)
Установление минимального срока действия пароля	Препятствует попыткам пользователя заменить пароль на старый после его смены по предыдущему требованию
Ведение журнала истории паролей	Обеспечивает дополнительную степень защиты по предыдущему требованию
Применение эвристического алгоритма, бракующего пароли на основании данных журнала истории	Усложняет задачу злоумышленника при попытке подобрать пароль по словарю или с использованием эвристического алгоритма
Ограничение числа попыток ввода пароля	Препятствует интерактивному подбору паролей злоумышленником
Поддержка режима принудительной смены пароля пользователя	Обеспечивает эффективность требования, ограничивающего максимальный срок действия пароля
Использование задержки при вводе неправильного пароля	Препятствует интерактивному подбору паролей злоумышленником
Запрет на выбор пароля самими пользователями и автоматическая генерация паролей	Исключает возможность подобрать пароль по словарю. Если алгоритм генерации паролей не известен злоумышленнику, последний может подбирать пароли только методом «тотального опробования»

Принудительная смена пароля при первой регистрации пользователя в системе	Защищает от неправомерных действия системного администратора, имеющего доступ к паролю в момент создания учетной записи

2. Примеры.

Например 1.

Задание определить время перебора всех паролей, состоящих из 6 цифр.

Алфавит составляют цифры n=10.

Длина пароля 6 символов k=6.

Таким образом, получаем количество вариантов: C=n=1 06

Примем скорость перебора X10 паролей в секунду. Получаем время перебора всех паролей t= C/б10^5секунд~1667минут~28часов~1,2 дня.

Примем, что после каждого из т=3 неправильно введенных паролей идет пауза в v=5 секунд. Получаем время перебора всех паролей

T=t*5/3=16667секунд~2778минут~46часов 1,9 дня.

^{T}итог = 1+T = 1,2 + 1,9 = 3,1 ДНЯ

2. Пример 2.

Определить минимальную длину пароля, алфавит которого состоит из 10 символов, время перебора которого было не меньше 10 лет.

Алфавит составляют символы n=10.

Длина пароля рассчитывается: k=1 оуп C=1 g C.

Определим количество вариантов C= t * б=10лет* 10 паролей в сек. = 10* 10* 365* 24* 60* 603,15* 10^9 вариантов

Таким образом, получаем длину пароля: k=1 g (3,15* 10^9) = 9,5

Очевидно, что длина пароля должна быть не менее 10 символов.

3. Задания..

1. Определить время перебора всех паролей с параметрами.

Алфавит состоит из n символов.

Длина пароля символов k.

Скорость перебора s паролей в секунду.

После каждого из т неправильно введенных паролей идет пауза в v секунд

вариант	n	k	s	т	v
1	33	10	100	o	o
2	2 6	1 2	1 3	3	2
3	52	6	30	5	10
4	66	7	20	10	3
5	59	5	200	0	0
6	118	9	50	7	12
7	128	10	500	0	0
8	150	3	200	5	3
9	250	8	600	7	3
10	500	5	1000	10	10

2. Определить минимальную длину пароля, алфавит которого состоит из П символов, время перебора которого было не меньше t лет.

Скорость перебора s паролей в секунду.

вариант	П	t	s
1	33	100	100
2	26	120	13
3	52	60	30
4	66	70	20
5	59	50	200
6	118	90	50
7	128	100	500
8	150	30	200
9	250	80	600
10	500	50	1000

3. Определить количество символов алфавита, пароль состоит из k символов, время перебора которого было не меньше t лет.

Скорость перебора s паролей в секунду.

вариант	k	t	s
1	5	100	100
2	6	120	13
3	10	60	30
4	7	70	20
5	9	50	200
6	11	90	50
7	12	100	500
8	6	30	200
9	8	80	600
10	50	50	1000

Контрольные вопросы:

1) Перечислить виды атак на пароли.
2) Перечислить критерии стойкости парольной защиты.
3) Перечислить и охарактеризовать методы противостояния атаке полным перебором.
4) Охарактеризовать влияние длины пароля на вероятность раскрытия.
5) Сформировать рекомендации по составлению паролей.
6) Перечислить типы угроз безопасности парольных систем.
7) Определить минимальную длину пароля, алфавит которого состоит из 10 символов, время перебора которого было не меньше 10 лет.
8) Определить время перебора всех паролей, состоящих из 6 цифр

Практическая работа №2
по учебной дисциплине **«Защита информации в системах управления»**

Раздел 1. Информационная безопасность информационных систем

Криптографическая защита информации.

1. Цель работы

Изучить мероприятия по созданию, распределению и хранению ключей при использовании пакета КРИПТОН® Подпись для организации электронного документооборота. Получить представление о задачах, возлагаемых на администратора и пользователей системы электронных документов.

2. Подготовка к занятию

1. Изучить (повторить) теоретический материал,.
2. Ознакомиться с программой лабораторной работы.
3. Подготовить отчет о лабораторной работе.
4. Ответить на контрольные вопросы.

3. Распределение времени занятия:

Всего: 90 мин
Вступительная часть 2 мин
Проверка готовности студентов к занятию 5 мин
Основная часть – 70 мин
Проверка выполнения практического занятия 10 мин
Заключительная часть 3 мин

4. Правила работы в лаборатории

К работе в лаборатории допускаются лица, изучившие правила и меры безопасности, сдавшие зачет по ним и усвоившие порядок выполнения лабораторной работы.

4.1. Требования безопасности перед началом работ

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить,

принимать пищу в лаборатории.

- Убедиться в целостности электрических розеток и разъемов. В лаборатории необходимо быть в сменной обуви.

- Включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

4.2. Требования безопасности во время работы

- выполняя практическое занятие, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и носители информации, непосредственно относящиеся к данному лабораторному занятию;

- подключение и отключение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса;

- при обнаружении неисправностей в оборудовании немедленно отключить источники питания и доложить об этом руководителю занятий или лаборанту.

4.3. Требования безопасности по окончании работы

- доложить руководителю занятий или лаборанту о завершении работ;

- привести в порядок и сдать рабочее место лаборанту, и доложить руководству

2.Порядок работы

Последовательно, в течение отведенного расписанием занятий времени, отработать следующие вопросы:

- изучить теоретический материал;
- оформить конспект к работе, получить допуск к выполнению работы;
- выполнить упражнения из практической части работы;

– оформить отчет по лабораторной работе и защитить его.

3.Теоретическое введение

Организация IETF разработала протокол для обеспечения безопасности в Internet, известный под названием IPSec. Протокол IPSec предназначен для криптографической защиты заголовка IP-пакета или всего IP-пакета. Этот протокол предусматривает обязательную аутентификацию IP-заголовка и возможную защиту информации об отправителе и адресате, содержащейся в некоторых полях IP-заголовка.

Поскольку данный протокол обеспечивает защиту информации на сетевом уровне, эта защита невидима для работающих приложений, т.е. он легко развертывается в существующих сетях.

Все протоколы, представленные в IPSec, можно разделить на протоколы непосредственно производящие обработку передаваемых данных (для обеспечения их защиты) и протоколы, которые позволяют автоматически согласовать параметры секретных соединений (далее – согласование), необходимые протоколам первой группы. Ядро IPSec составляют три протокола: протокол аутентификации (Authentication Header, AH), протокол шифрования (Encapsulating Security Payload, ESP) и протокол обмена ключами (Internet Key Exchange, IKE). Функции по поддержанию защищенного канала распределяются между этими протоколами следующим образом:

– протокол AH гарантирует целостность и аутентичность данных;

– протокол ESP шифрует передаваемые данные, гарантируя конфиденциальность, но он может также поддерживать аутентификацию и целостность данных;

– протокол IKE решает вспомогательную задачу автоматического предоставления конечным точкам канала секретных ключей, необходимых для работы протоколов аутентификации и шифрования данных.

IPSec обеспечивает проверку подлинности на уровне компьютера и шифрование данных для VPN-подключений, использующих протокол L2TP. IPSec выполняет согласование между компьютером и удаленным сервером туннеля перед установлением L2TP-подключения, что защищает и пароли, и данные.

L2TP использует с IPSec стандартные протоколы проверки подлинности PPP, такие как EAP, MS-CHAP, CHAP, SPAP и PAP.

Уровень шифрования определяется параметрами секретных соединений, называемых также сопоставлением безопасности IPSec. Сопоставление безопасности – это комбинация адреса назначения, протокола безопасности и уникального идентификатора, называемого индексом параметров безопасности.

Протоколы безопасности IPSEC

Протоколы безопасности обеспечивают защиту данных и подлинности для каждого пакета IP. Служба IPSEC в Windows 2000 использует протоколы AH и EPS.

AH (Authentication Header)

Протокол AH обеспечивает проверку подлинности, целостность и отсутствие повторов для всего пакета (заголовка IP и полезных данных); AH подписывает весь пакет. Данные не шифруются, поэтому конфиденциальность не обеспечивается. Данные доступны для чтения, но защищены от изменения. Протокол AH использует для подписания пакетов алгоритмы HMAC.

Например, Мария, работающая на компьютере A, отправляет данные Ивану на компьютер B. Заголовок IP, заголовок AH и данные защищены от изменения подписью. Это позволяет Ивану быть уверенным в том, что данные были отправлены именно Марией и не были изменены.

Проверка целостности и подлинности обеспечивается путем вставки заголовка AH между заголовком IP и заголовком транспортного протокола (TCP или UDP).

ESP (Encapsulating Security Payload)

ESP помимо проверки подлинности, целостности и отсутствия повторов обеспечивает также конфиденциальность.

ESP обычно не подписывает пакеты, если не используется туннель. Обычно обеспечивается только защита данных, но не заголовка IP.

Например, Мария, работающая на компьютере A, отправляет данные Ивану на компьютер B. Данные шифруются, поскольку ESP обеспечивает конфиденциальность. При получении, после завершения процесса проверки, данные пакета расшифровываются. Иван может быть уверен в том, что данные отправлены именно Марией, что они не были изменены, а также в том, что никто другой не сможет их прочесть.

Безопасность обеспечивается путем вставки заголовка ESP между и заголовком транспортного протокола (TCP или UDP).

Протокол IKE

Протокол IKE представляет собой набор протоколов аутентификации и обмена аутентифицированными ключами. Каждый протокол из этого набора представляет собой гибрид, использующий часть протокола Окли (протокол определения ключа Окли - Oakley Key Determination Protocol), часть механизма SKEME (Механизм для многостороннего обмена секретными ключами через Internet – Versatile Secure Key Exchange Mechanism for Internet) и часть протокола ISAKMP (Протокол установления защищенных соединений и управления ключами-Internet Security Association and Key Management Protocol).

Протокол ISAKMP обеспечивает аутентификацию и обмен аутентифицированными ключами между двумя общающимися сторонами, позволяя им согласовывать и утверждать различные защитные средства, криптографические алгоритмы, параметры безопасности, механизмы аутентификации и тому подобные, т.е. все то, что в совокупности называется сопоставлением безопасности (secure associations). Однако в протоколе ISAKMP не определены конкретные способы обмена ключами, что позволяет

применять его в сочетании с разными протоколами.

Будучи гибридом, протокол IKE представляет собой набор двусторонних протоколов, предназначенных для обмена аутентифицированными сеансовыми ключами, большинство из которых использует механизм обмена ключами Диффи-Хеллмана и предоставляет пользователям широкие возможности для переговоров в интерактивном режиме.

Принцип работы IPSEC

Для наглядности в данном примере рассматривается использование IPSEC для компьютеров одного домена. Алиса, работающая с приложением на компьютере A, отправляет сообщение Бобу.

1. Драйвер IPSEC на компьютере A проверяет список фильтров IP в активной политике на наличие совпадающего адреса или типа трафика исходящих пакетов.
2. Драйвер IPSEC предоставляет ISAKMP сведения для начала согласования безопасности с компьютером B.
3. Служба ISAKMP на компьютере B получает запрос для согласования безопасности.
4. Два компьютера выполняют обмен ключами, устанавливают соответствие безопасности ISAKMP и создают общий секретный ключ.
5. Два компьютера согласовывают уровень безопасности для передачи данных, устанавливая пару

соответствий безопасности IPSEC и ключей для защиты пакетов IP.

6. Используя сопоставление безопасности IPSEC для исходящего трафика и ключ, драйвер IPSEC на компьютере A подписывает пакеты для проверки целостности и зашифровывает пакеты, если было согласовано шифрование.

7. Драйвер IPSEC на компьютере A отправляет пакеты на соответствующий тип подключения для передачи на компьютер B.

8. Компьютер B получает защищенные пакеты и передает их драйверу IPSEC.

9. Используя сопоставление безопасности IPSEC для входящего трафика и ключ, драйвер IPSEC на компьютере B проверяет подпись целостности и, при необходимости, расшифровывает пакеты.

10. Драйвер IPSEC на компьютере B передает расшифрованные пакеты драйверу TCP/IP, который передает их в принимающее приложение.

Для Алисы и Боба все эти процессы не видны. Стандартные маршрутизаторы и коммутаторы, передающие данные между сторонами подключения, не требуют использования IPSEC. Они автоматически пересылают зашифрованные пакеты IP в место назначения. Однако, если маршрутизатор функционирует как брандмауэр, шлюз безопасности или прокси-сервер, для прохождения безопасных пакетов IP необходимо включить специальную фильтрацию.

Управление политикой безопасности IP

Оснастка «Управление политикой безопасности IP» используется для настройки в Windows IP-безопасности (IPSec), ключевой линии обороны на случай внутренних (частная сеть) и внешних (Интернет, внешние сети) атак. Хотя в большинстве стратегий сетевой безопасности акцент делается на

предотвращение атак на сеть организации извне, множество важных сведений может быть потеряно в результате внутренних атак с расшифровкой передаваемых по сети данных. Большая часть данных передается по сети в зашифрованном виде, но сотрудники, обслуживающий персонал или посетители могут подключиться к сети и скопировать данные для последующего анализа. Кроме того, могут быть предприняты атаки на другие компьютеры на сетевом уровне. Брандмауэры для защиты от такой внутренней угрозы не годятся, поэтому использование IPSEC значительно повышает безопасность корпоративных данных.

IPSEC шифрует данные, передаваемые между двумя компьютерами, защищая их от изменения и интерпретации при просмотре в сети. Администратор должен сначала определить доверительные отношения между двумя компьютерами, а затем указать для них способ защиты трафика. Эта конфигурация содержится в политике IPSEC, создаваемой и применяемой администратором.

Свойства политики IPSEC

Политики IPSEC могут применены к локальным компьютерам, членам домена, доменам, организационным подразделениям или любым объектам групповой политики в Active Directory. Политики IPSEC организации должны основываться на принятых в организации правилах безопасной работы. Политики могут содержать несколько действий безопасности, называемых правилами, что позволяет применять одну политику к нескольким компьютерам.

Для хранения политик IPSEC используются два расположения.

1. Active Directory.
2. Определен локально в реестре для автономных компьютеров и компьютеров, не являющихся частью доверенного домена Windows 2000/2003 постоянно. Если компьютер временно не подключен к доверенному домену Windows 2000/2003, сведения политики кэшируются в локальном реестре.

Windows содержит предопределенные политики, которые могут быть активизированы, изменены в соответствии с потребностями или использованы в качестве шаблона при создании собственных настраиваемых политик. Каждая определенная политика должна применяться к сценарию плана безопасности. При назначении политики серверу DHCP, DNS (Domain Name System), WINS, SNMP (Simple Network Management Protocol) или серверу удаленного доступа можно использовать дополнительные параметры.

Политики IPSEC, назначенные объекту групповой политики в Active Directory, становятся частью групповой политики и переносятся на компьютеры, входящие в Active Directory при распространении групповой политики.

При назначении политики IPSEC в Active Directory следует иметь в виду следующее.

- Политики IPSEC, назначенные политике домена, подавляют любую локальную активную политику IPSEC только в том случае, если компьютер является членом этого домена.
- Политики IPSEC, назначенные организационному подразделению, подавляют политику IPSEC, назначенную политике домена, на всех компьютерах, входящих в это организационное подразделение. Политика IPSEC, назначенная организационному подразделению низшего уровня, подавляет политику IPSEC, назначенную организационному подразделению высшего уровня на всех компьютерах, входящих в это подразделение.

Предопределенные политики IPSEC

Windows 2000/XP содержит набор предопределенных политик IPSEC. По умолчанию все предопределенные политики предназначены для компьютеров, являющихся членами домена Windows 2000/2003. Их можно назначать без дополнительной настройки, изменять или использовать в

качестве шаблонов при создании собственных политик

Предопределенные политики

Клиент (Только ответ)

Используется только для компьютеров, безопасная связь для которых большую часть времени не требуется. Например, клиенты интрасети могут не требовать использования IPSEC, кроме случаев, когда запрос исходит с другого компьютера. Эта политика позволяет компьютеру, на котором она активизирована, должным образом отвечать на запросы безопасной связи. Эта политика содержит стандартное правило ответа, позволяющее выполнять согласование с компьютерами, требующими использования IPSEC. Защита применяется только к трафику по запрошенному протоколу и через запрошенный порт.

Сервер (запрос безопасности)

Используется для компьютеров, для которых большую часть времени требуется безопасная связь. В качестве примера можно привести сервер, через который передаются важные данные. В данной политике компьютер принимает незащищенный трафик, но всегда выполняет попытку защитить дополнительные связи, посылая отправителю запрос безопасности. Эта политика допускает полное отсутствие защиты трафика, если другой компьютер не поддерживает IPSEC.

Безопасность сервера (требовать безопасность)

Эта политика используется для компьютеров, постоянно требующих безопасной связи. Примером может служить сервер, передающий весьма важные данные, или шлюз безопасности, защищающий интрасеть от посторонних. Эта политика отклоняет небезопасные входящие связи, а исходящий трафик всегда защищен. Небезопасная связь не допускается, даже если другая сторона не поддерживает IPSEC.

Предопределенные правила

Как и предопределенные политики, предопределенное правило отклика может быть активизировано без дополнительной настройки или изменено в

соответствии с конкретными потребностями. Оно добавляется в каждую создаваемую политику, но не активизируется автоматически. Правило предназначено для любых компьютеров, которые не требуют безопасности, но должны иметь возможность дать соответствующий отклик при запросе безопасной связи от другого компьютера.

Предопределенные действия фильтра

Как и предопределенные правила, предопределенные действия фильтра могут быть активизированы без дополнительной настройки, изменены или использованы в качестве шаблона при определении собственных действий фильтра. Следующие действия доступны для активизации в любом новом или существующем правиле.

- **Требовать безопасность**. Высокая безопасность. Небезопасные связи не допускаются.
- **Запрос безопасности (необязательно).** Безопасность от средней до низкой. Небезопасная связь допускается для обеспечения связи с компьютерами, не выполняющими согласование IPSEC.

Добавление или изменение политики IPSEC

1. В оснастке «Управление политикой безопасности IP» выберите создание новой политики или изменение текущей.

Чтобы	**Выполните следующие действия**
Создать новую политику	Выберите в дереве консоли **Политики безопасности IP на** описание, а затем выберите в меню **Действие** команду **Создать политику безопасности IP.** Выполняйте инструкции мастера политики безопасности IP до вывода на экран диалогового окна новой политики.
Изменить существующую политику	Щелкните нужную политику правой кнопкой мыши и выберите команду **Свойства.**

2. На вкладке **Общие** введите уникальное имя в поле **Имя**.

3. В поле **Описание** введите описание политики безопасности, например укажите группы или домены, которые она затрагивает.

4. Если компьютер является частью домена, введите значение в поле **Проверять политику на наличие изменений каждые** число **минут**, чтобы задать частоту выполнения агентом политики проверки групповой политики на наличие обновлений

5. При наличии особых потребностей в безопасности при обмене ключами нажмите кнопку **Дополнительно**.

6. Перейдите на вкладку **Правила** и создайте все необходимые правила для политики.

Добавление и изменение правила

1. В оснастке «Управление политикой безопасности IP» щелкните правой кнопкой политику, которую требуется изменить, и выберите команду **Свойства**.

2. Выберите, следует ли использовать мастер добавления:

— чтобы использовать мастер, помогающий последовательно выполнить операции по добавлению правила безопасности, убедитесь в том, что флажок **Использовать мастер** установлен, нажмите кнопку **Далее** и следуйте выводимым на экран инструкциям;

— для того чтобы добавить или изменить правило вручную, снимите флажок **Использовать мастер**, нажмите кнопку **Добавить** или **Изменить** и выполните следующие шаги данной процедуры.

3. На соответствующих вкладках задайте список фильтров IP, действие фильтра, тип подключения, методы проверки подлинности и параметры туннеля.

Добавление и изменение действия фильтра

1. В оснастке «Управление политикой безопасности IP» щелкните правой кнопкой политику, которую требуется изменить, и выберите команду **Свойства**.

2. Выберите правило, которое требуется изменить, нажмите кнопку **Изменить**, а затем перейдите на вкладку **Действие фильтра**.

3. Выберите, следует ли использовать мастер добавления:

— чтобы использовать мастер, помогающий последовательно выполнить операции по добавлению действия фильтра, убедитесь в том, что флажок **Использовать мастер** установлен, а затем следуйте выводимым на экран инструкциям;

— для того чтобы выполнить добавление или изменение действий фильтра вручную, снимите флажок **Использовать мастер**, а затем нажмите кнопку **Добавить** для добавления действия фильтра или кнопку **Изменить** для изменения настроек действия фильтра.

4. Выберите действие фильтра.

— Выберите **Разрешить**, чтобы разрешить получение или отправку незашифрованных пакетов. Безопасность для этих пакетов запрашиваться не будет.

— Выберите **Блокировать**, чтобы пакеты, совпадающие с данным фильтром, немедленно отбрасывались. Безопасность для этих пакетов запрашиваться не будет.

—

— Выберите **Согласовать безопасность**, чтобы использовать для обеспечения безопасности пакетов, совпадающих с фильтром, методы безопасности из списка **Методы безопасности в порядке предпочтения**. Запросы безопасности для этих пакетов будут приниматься.

5. Если входящие незащищенные пакеты блокировать не требуется, но необходимо обеспечить безопасность исходящих запросов и последующего двустороннего обмена данными, установите флажок **Принимать небезопасную связь, но отвечать с помощью IPSEC**.

6. Если требуется сделать возможной связь с другими компьютерами, не поддерживающими IPSEC, и обеспечить продолжение связи при отсутствии ответа на запрос согласования безопасности IPSEC,

установите флажок **Разрешать связь с компьютерами, не поддерживающими IPSEC**. Если этот параметр включен, то после сбоя согласования IPSEC с конкретным узлом безопасность IPSEC для связи с этим узлом отключается на некоторое время.

7. Установка флажка **Сеансовые циклы безопасной пересылки (PFS)** гарантирует, что основные ключи или исходные данные ключа не используются для создания ключа сеанса более одного раза.

8. На вкладке **Общие** введите уникальное имя.

9. Введите описание. Например, можно указать уровни безопасности, представляемые данным действием ключа.

10. Если выбрано действие **Согласовать безопасность**, добавьте новые или измените существующие методы безопасности для действия фильтра.

Задание туннеля IPSEC

1. В области сведений оснастки «Управление политикой безопасности IP» щелкните правой кнопкой политику, которую требуется изменить, и выберите команду **Свойства**.

2. Выберите правило, которое требуется изменить, и нажмите кнопку **Изменить.**

3. На вкладке **Параметры туннеля** укажите компьютер, который будет конечной точкой туннеля:

Чтобы	**Выполните следующие действия**
Отключить туннелирование для данного правила	Выберите **Это правило не указывает туннель IPSEC**.
Использовать туннелированную связь с конкретным компьютером	Выберите **Конечная точка туннеля указана данным IP-адресом**.

Определение методов проверки подлинности IPSEC

1. В оснастке «Управление политикой безопасности IP» щелкните правой кнопкой политику, которую требуется изменить, и выберите команду **Свойства**.

2. Выберите правило, которое требуется изменить, и нажмите кнопку **Изменить.**

3. На вкладке **Методы проверки подлинности** нажмите кнопку **Добавить**. При изменении настроек существующего метода выберите изменяемый метод проверки подлинности и нажмите кнопку **Изменить.**

4. Определите методы проверки подлинности:

— выберите **Стандарт Windows 2000 (Kerberos V5)**, чтобы использовать для служб проверки подлинности протокол безопасности Kerberos V5, если данное правило применяется к компьютерам, которые заверены доверенным доменом Windows 2000;

— чтобы использовать для служб проверки подлинности сертификат открытого ключа, выберите **Использовать сертификат данного Центра сертификации**.

— Если выбрано использование сертификата, нажмите кнопку **Обзор** для выбора корневого центра сертификации.

— Чтобы указать для использования при проверке подлинности собственный ключ, выберите **Использовать данную строку для защиты обмена ключами**.

Указание типа подключения IPSEC

1. В оснастке «Управление политикой безопасности IP» щелкните правой кнопкой политику, которую требуется изменить, и выберите команду **Свойства**.

2. Выберите правило, которое требуется изменить, и нажмите кнопку **Изменить.**

3. На вкладке **Тип подключения** выберите тип сетевых подключений, для которых будет применяться данное правило:

— чтобы применить правило ко всем сетевым подключениям, созданным на данном компьютере, выберите **Все сетевые подключения**;

— чтобы применить правило ко всем локальным подключениям, созданным на данном компьютере, выберите **Локальное сетевое подключение**;

— чтобы применить правило к удаленным сетевым подключениям, созданным на данном компьютере, выберите **Удаленный доступ**.

4.Практическая часть

Создать политику безопасности IPSEC, реализующей следующие требования:

1. Весь SMTP и POP3-трафик между локальным компьютером и любым другим, поддерживающим шифрование, должен использовать обязательное шифрование для всех сетевых подключения.
2. Весь IP-трафик между локальным компьютером и подсетью 195.17.72.0/255.255.255.0 должен использовать обязательное шифрование для всех сетевых подключения.
3. Весь IP-трафик между локальным компьютером и подсетью 195.17.73.0/255.255.255.0 должен использовать обязательное шифрование для всех сетевых подключения с использованием сертификата открытого ключа для проверки подлинности, выданного центром сертификации VeriSign.

Отчет должен содержать последовательность действий по созданию и настройке параметров политики, правил и фильтров.

5.Контрольные вопросы

1. Назначение протокола IPSec.
2. Состав семейства протоколов IPSec.
3. Средства настройки IPSec в Windows 2000/XP.

4. Состав политики безопасности.
5. Создание правил и фильтров для политики безопасности.
6. Возможные методы проверки подлинности IPSec.
7. Возможные места хранения назначенных политик безопасности.

Основная литература

1. Соколов А.В., Шаньгин В.Ф. Защита информации в распределенных корпоративных сетях и системах. М.: ДМК Пресс, 2005

2. Мао Венбо Современная криптография: теория и практика. М.: Издательский дом «Вильямс», 2005

Дополнительная литература

1. Справочная система ОС Windows 2000/XP

Практическая работа №3
по учебной дисциплине **«Защита информации в системах управления»**

Раздел 1. Информационная безопасность информационных систем

Межсетевое экранирование.

1. Цель работы:

Изучение механизмов работы средств обеспечения и поддержки сетевой защиты – брандмауэра и сетевого сканера;

Практическое ознакомление с работой сетевого сканера XSpider и межсетевого экрана «Outpost»;

Изучение работы рассматриваемых систем в «совместном» режиме для защиты и тестирования рабочих станций на наличие уязвимостей.

2. Подготовка к занятию

1. Изучить (повторить) теоретический материал.
2. Ознакомиться с программой лабораторной работы.
3. Подготовить отчет о лабораторной работе.
4. Ответить на контрольные вопросы.

3. Распределение времени занятия:

Всего: 90 мин

Вступительная часть 2 мин

Проверка готовности студентов к занятию 5 мин

Основная часть – 70 мин

Проверка выполнения практического занятия 10 мин

Заключительная часть 3 мин

4. Правила работы в лаборатории

К работе в лаборатории допускаются лица, изучившие правила и меры безопасности, сдавшие зачет по ним и усвоившие порядок выполнения лабораторной работы.

4.1. Требования безопасности перед началом работ

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории.

- Убедиться в целостности электрических розеток и разъемов. В лаборатории необходимо быть в сменной обуви.

- Включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

4.2. Требования безопасности во время работы

- выполняя практическое занятие, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и носители информации, непосредственно относящиеся к данному лабораторному занятию;

- подключение и отключение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса;

- при обнаружении неисправностей в оборудовании немедленно отключить источники питания и доложить об этом руководителю занятий или лаборанту.

4.3. Требования безопасности по окончании работы

- доложить руководителю занятий или лаборанту о завершении работ;

- привести в порядок и сдать рабочее место лаборанту, и доложить руководству

1. Теоретические сведения.

Интенсивная информатизация государственных и муниципальных управленческих структур, промышленных предприятий и корпораций, силовых ведомств, научных, медицинских и других учреждений выдвинула на первый план вопросы безопасности информационных ресурсов.

Среди угроз безопасности информации значительное место занимает автоматическое внедрение в компьютеры программных закладок, способных скрыто отслеживать и передавать злоумышленнику данные о функционировании компьютера, обрабатываемой на нем информации, а также о всей компании в целом. Кроме того, проблемы компьютерным сетям предприятий создают факты проникновения компьютерных хулиганов, которые, взломав систему сетевой защиты компании, могут завладеть конфиденциальной информацией или нанести физический вред оборудованию, используя специализированное вредоносное ПО.

Подобные ситуации возникают из-за уязвимостей в системе корпоративной защиты компании, в основном связанные с открытыми портами неиспользуемых сервисов, работающих вхолостую. Как правило, через «дыры» в данных сервисах осуществляется большая часть удачных атак извне, которые, зачастую, кончаются потерей компанией секретных данных.

Ярким примером наличия подобных уязвимостей могут послужить популярные операционные системы WindowsXP и FreeBSD. Так, в MS Windows, по-умолчанию, работает довольно много неиспользуемых сервисов, которые в большинстве своем связаны с открытыми портами, через которые злоумышленник может провести атаку. Всем, наверное, известен факт, когда множество «автономных» пользовательских компьютеров пострадали во всем мире в результате атаки на порт 135 (RPC). Что касается FreeBSD, то здесь также после стандартной установки в системе работают демоны, которые в обычных случаях не требуются, а значит, являются дополнительными источниками уязвимостей в компьютере. Атаки на почтовый сервер sendmail приводят к полному получению злоумышленником контроля над хостом. Откуда sendmail, спросите вы? Да, иногда, в UNIX-системах, в том числе адаптированных для работы в качестве рабочей станции, в конфигурации «по-умолчанию» можно встретить и такие сервисы…

Необходимо отметить, что на сегодняшний день работы по проникновению злоумышленников через «дыры» в защите на 90% автоматизированы. Поэтому, «самостоятельное» появление вредоносного ПО на вашем компьютере, которое встречается сегодня очень часто, связано в большинстве случаев с наличием непреднамеренных лазеек в неиспользуемых, а значит, не обновляющихся, службах.

Тем не менее, при использовании специальных средств защиты, подобных нежелательных событий можно, как правило, избежать.

Основными средствами защиты на сегодняшней день являются две категории специализированных программ:

— Межсетевые экраны (брандмауэры, FireWall, МСЭ);

— Сканеры (сканеры открытых портов и сервисов).

Следует сказать, что брандмауэр – основной механизм в сети программной и аппаратной защиты рабочих станций и серверов от атак извне и изнутри.

Сканер – это вспомогательный программный инструмент, позволяющий провести групповое тестирование параметров хостов сети, а также определить наличие и правильность настройки в них МСЭ.

Эти два класса систем в комплексе позволяют построить эффективную эшелонированную систему защиты компании, значительно снизив тем самым вероятность вторжения в сеть злоумышленников.

1.1. Системы программной и аппаратной защиты рабочих станций – брандмауэры (FireWalls).

Архитектура firewall

Firewall — это шлюз сети, снабженный правилами защиты. Он может быть аппаратным или программным. В соответствии с заложенными правилами обрабатывается каждый пакет, проходящий наружу или внутрь сети, причем процедура обработки может быть задана для каждого правила. Производители программ и машин, реализующих firewall-технологии, обеспечивают различные способы задания правил и процедур. Обычно firewall создает контрольные записи, детализирующие причину и обстоятельства возникновения внештатных ситуаций. Анализируя такие контрольные записи, администраторы часто могут обнаружить источники атаки и способы ее проведения.

Фильтрация пакетов (packet filtering firewalls)

Каждый IP-пакет проверяется на совпадение заложенной в нем информации с допустимыми правилами, записанными в firewall.

Параметры, которые могут проверяться:

— физический интерфейс движения пакета;
— адрес, с которого пришел пакет (источник);
— адрес, куда идет пакет (получатель);
— тип пакета (TCP, UDP, ICMP);
— порт источника;
— порт получателя.

Механизм фильтрации пакетов не имеет дела с их содержанием. Это позволяет использовать непосредственно ядро операционной системы для задания правил. В сущности, создаются два списка: отрицание (deny) и разрешение (permit). Все пакеты должны пройти проверку по всем пунктам этого списка. Далее используются следующие методы:

— если никакое правило соответствия не найдено, то удалить пакет из сети;
— если соответствующее правило найдено в списке разрешений, то пропустить пакет;
— если соответствующее правило найдено в списке отрицаний, то удалить пакет из сети.

В дополнение к этому firewall, основанный на фильтрации пакетов, может изменять адреса источников пакетов, выходящих наружу, чтобы скрыть тем самым топологию сети (метод address translation), плюс осуществляет условное и безусловное перенаправление пакетов на другие хосты. Отметим преимущества firewall, основанного на фильтрации пакетов:

— фильтрация пакетов работает быстрее других firewall-технологий, потому что используется меньшее количество проверок;
— этот метод легко реализуем аппаратно;
— одно-единственное правило может стать ключевым при защите всей сети;
— фильтры не требуют специальной конфигурации компьютера;
— метод address translation позволяет скрыть реальные адреса компьютеров в сети.

Однако имеются и недостатки:

— нет проверки содержимого пакетов, что не дает возможности, например, контролировать, что передается по FTP. В этом смысле application layer и circuit level firewall гораздо практичнее;
— нет информации о том, какой процесс или программа работали с этим пакетом, и сведений о сессии работы;
— работа ведется с ограниченной информацией пакета;
— в силу «низкоуровневости» метода не учитывается особенность передаваемых данных;
— слабо защищен сам компьютер, на котором запущен firewall, то есть предметом атаки может стать сам этот компьютер;
— нет возможности сигнализировать о внештатных ситуациях или выполнять при их возникновении какие-либо действия;
— возможно, что большой объем правил будет тормозить проверку.

Firewall цепного уровня (circuit level firewalls)

Поскольку при передаче большой порции информации она разбивается на маленькие пакеты, целый фрагмент состоит из нескольких пакетов (из цепи пакетов). Firewall цепного уровня проверяет целостность всей цепи, а также то, что она вся идет от одного источника к одному получателю, и информация о цепи внутри пакетов (а она там есть при использовании TCP)

совпадает с реально проходящими пакетами. Причем цепь вначале собирается на компьютере, где установлен firewall, а затем отправляется получателю. Поскольку первый пакет цепи содержит информацию о всей цепи, то при попадании первого пакета создается таблица, которая удаляется лишь после полного прохождения цепи. Содержание таблицы следующее:

- уникальный идентификатор сессии передачи, который используется для контроля;
- состояние сессии передачи: установлено, передано или закрыто;
- информация о последовательности пакетов;
- адрес источника цепи;
- адрес получателя цепи;
- физический интерфейс, используемый для получения цепи;
- физический интерфейс, используемый для отправления цепи.

Эта информация применяется для проверки допустимости передачи цепи. Правила проверки, как и в случае фильтрации пакетов, задаются в виде таблиц в ядре. Основные преимущества firewall цепного уровня:

- firewall цепного уровня быстрее программного, так как производит меньше проверок;
- firewall цепного уровня позволяет легко защитить сеть, запрещая соединения между определенными адресами внешней и внутренней сети;
- возможно скрытие внутренней топологии сети.

Недостатки firewall цепного уровня:

- нет проверки пакетов на программном уровне;
- слабые возможности записи информации о нештатных ситуациях, кроме информации о сессии передачи;
- нет проверки передаваемых данных;

— трудно проверить разрешение или отрицание передачи пакетов.

Firewall программного уровня

Помимо целостности цепей, правильности адресов и портов, проверяются также сами данные, передаваемые в пакетах. Это позволяет проверять целостность данных и отслеживать передачу таких сведений, как пароли. Вместе с firewall программного уровня используется proxy-сервис, который кэширует информацию для более быстрой ее обработки. При этом возникают такие новые возможности, как, например, фильтрация URL и установление подлинности пользователей. Все соединения внутренней сети с внешним миром происходят через proxy, который является шлюзом. У proxy две части: сервер и клиент. Сервер принимает запросы, например на telnet-соединение из внутренней сети с внешней, обрабатывает их, то есть проверяет на допустимость передачи данных, а клиент работает с внешним компьютером от имени реального клиента. Естественно, вначале все пакеты проходят проверку на нижних уровнях. Достоинства proxy:

— понимает и обрабатывает протоколы высокого уровня типа HTTP и FTP;
— сохраняет полную информацию о сессии передачи данных как низкого, так и высокого уровня;
— возможен запрет доступа к некоторым сетевым сервисам;
— есть возможность управления пакетами данных;
— есть сокрытие внутренних адресов и топологии сети, так как proxy является фильтром;
— остается видимость прямого соединения сетей;
— proxy может перенаправлять запросы сетевых сервисов на другие компьютеры;
— есть возможность кэширования http-объектов, фильтрации URL и установления подлинности пользователей;

— возможно создание подробных отчетных записей для администратора.

Недостатки proxy:

— требует изменения сетевого стека на машине, где стоит firewall;
— нельзя напрямую запустить сетевые сервисы на машине, где стоит firewall, так как proxy перехватывает работу портов;
— неминуемо замедляет работу, потому все данные обрабатываются дважды: «родной» программой и собственно proxy;
— так как proxy должен уметь работать с данными какой-либо программы, то для каждой программы нужен свой proxy;
— нет proxy для UDP и RPC;
— иногда необходима специальная настройка клиента для работы с proxy;
— proxy не защищен от ошибок в самой системе, а его работа сильно зависит от наличия последних;
— корректность работы proxy напрямую связана с правильностью обработки сетевого стека;
— использование proxy может требовать дополнительных паролей, что неудобно для пользователей.

Динамическая фильтрация пакетов (dynamic packet filter firewalls)

В основном этот уровень повторяет предыдущий, за двумя важными исключениями:

— возможно изменение правил обработки пакетов «на лету»;
— включена поддержка UDP.

Уровень kernel proxy

Уровень kernel proxy возник достаточно недавно. Основная его идея — попытка поместить описанный выше алгоритм firewall программного уровня в ядро операционной системы, что избавляет компьютер от лишних затрат

времени на передачу данных между ядром и программой proxy. Это повышает производительность и позволяет производить более полную проверку проходящей информации.

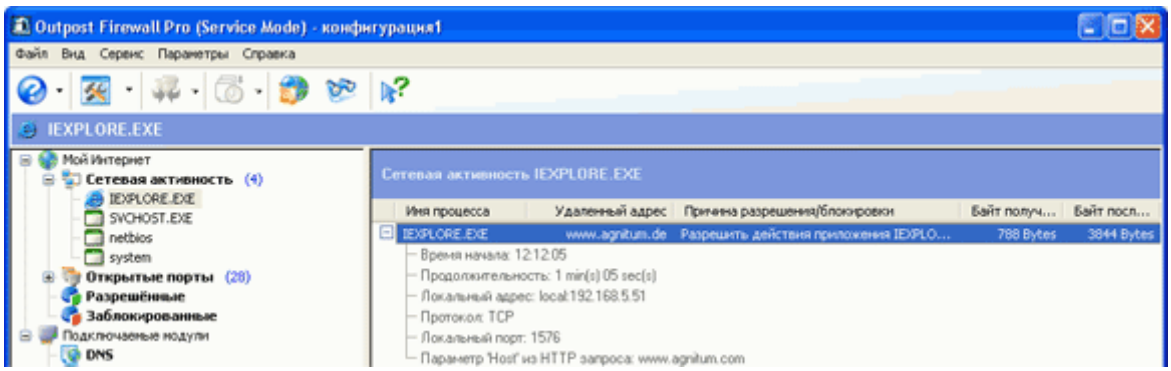
Примеры межсетевых экранов

1. Аппаратный (D-Link)

DFL-1100

Межсетевой экран для сетей крупных предприятий

2. Программный (Agnitum Outpost)



1.2. Вспомогательные системы обеспечения безопасности компьютерных сетей - сканеры.

Архитектура сканера

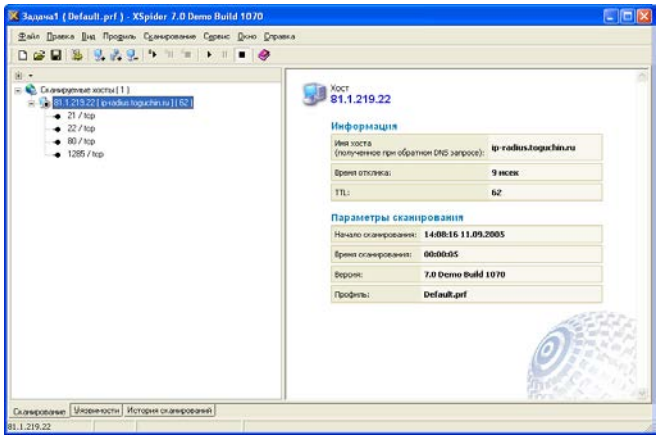
Основной принцип функционирования сканера заключается в эмуляции действий потенциального злоумышленника по осуществлению сетевых атак. Поиск уязвимостей путем имитации возможных атак является одним из наиболее эффективных способов анализа защищенности АС, который дополняет результаты анализа конфигурации по шаблонам, выполняемый локально с использованием шаблонов (списков проверки). Современные сканеры способны обнаруживать сотни уязвимостей сетевых ресурсов, предоставляющих те или иные виды сетевых сервисов. Их предшественниками считаются сканеры телефонных номеров (war dialers), использовавшиеся с начала 80-х и непотерявшие актуальности по сей день. Первые сетевые сканеры представляли собой простейшие сценарии на языке Shell, сканировавшие различные TCP-порты. Сегодня они превратились в зрелые программные продукты, реализующие множество различных сценариев сканирования. Современный сетевой сканер выполняет четыре основные задачи:

— Идентификацию доступных сетевых ресурсов;
— Идентификацию доступных сетевых сервисов;
— Идентификацию имеющихся уязвимостей сетевых сервисов;
— Выдачу рекомендаций по устранению уязвимостей.

В функциональность сетевого сканера не входит выдача рекомендаций по использованию найденных уязвимостей для реализации атак на сетевые ресурсы. Возможности сканера по анализу уязвимостей ограничены той информацией, которую могут предоставить ему доступные сетевые сервисы. Принцип работы сканера заключается в моделировании действий злоумышленника, производящего анализ сети при помощи стандартных сетевых утилит, таких как host, showmount, traceout, rusers, finger, ping и т. п. При этом используются известные уязвимости сетевых сервисов, сетевых протоколов и ОС для осуществления удаленных атак на системные ресурсы и осуществляется документирование удачных попыток.

Число уязвимостей в базах данных современных сканеров медленно, но уверенно приближается к 10000.

Пример сканера XSpider



2. Порядок выполнения работы.

Условия выполнения лабораторной работы. Данная работа должна выполняться в присутствии администратора компьютерного класса или уполномоченного им лица, которому предоставляются права на осуществление следующих действий в операционных системах Windows2000 – XP, работающих как в сетевом режиме, так и в одиночном режиме:

1.1. права на установку программного обеспечения;

1.2. права на работу как в составе рабочей группы или домена Windows, так и в составе локального администратора рабочей станции;

1.3. права на предоставление учетной записи учащимся, позволяющей установку ПО.

Лабораторная работа проводится в двух вариантах:

1. Автономный.

В компьютерном классе должны находиться не менее 2-х машин, объединенных в сеть. На первой устанавливается *сетевой сканер* или *межсетевой экран*. В случае установки МСЭ вторая машина используется

для тестирования защиты первой от ICMP пакетов с помощью стандартной утилиты *ping*. При «автономном» тестировании сканера вторая машина будет использоваться в качестве исследуемого объекта.

2. Совместный.

В компьютерном классе должны находиться также не менее 2-х машин, объединенных в сеть. На части из них устанавливается *сканер*, на остальных – МСЭ. При этом для проверки защиты рабочей станции от ICMP пакетов с помощью МСЭ (а также для тестирования сканера) будет использоваться сетевой сканер.

Администратор! Обрати внимание. После установки изучаемого ПО межсетевые экраны могут заблокировать доступ сетевого трафика к рабочим станциям, тем самым, нарушив работоспособность сети. Для предотвращения данной ситуации необходимо сразу назначить всем МСЭ политику «разрешения».

Порядок работы

Работа будет проходить в два этапа. Первый этап предназначен для изучения работы XSpider, Outpost и WindowsXP в «автономном» режиме. Второй этап – для изучения работы в совместном режиме. На каждом этапе студенты делятся на две группы, одна из которых будет работать с МСЭ, вторая – со сканером или псевдосканером (утилитой ping).

Действия, общие для двух этапов:

1. Взять из папки \\m00\fit2005 файлы установки сканера XSpider и МСЭ Agnitum Outpost;
2. На каждом рабочем месте выполнить установку сканера и МСЭ;
3. При установке Outpost соглашаться со всеми вопросами. По окончании установки – перезагрузить машину;

4. Перед началом работы перевести установленный МСЭ в режим разрешения. Открыть Outpost -> меню «Параметры» -> «Политики» -> выбрать режим «Разрешать»;
5. Узнать имя и IP-адрес своего рабочего компьютера: «Пуск» -> «Выполнить» -> “cmd” -> “ipconfig /all”;

Этап 1. Автономный режим. Каждый студент из группы 1 должен работать в паре со студентом из группы 2.

Группа 1:

1. Запустить пинг компьютера-соседа из группы 2: «Пуск» -> «Выполнить» -> “cmd” -> “ping ip-addr -t”; (утилита ping располагается в C:\windows\system32)
2. Смотреть на ответные пинг-пакеты.
3. Фиксировать моменты, когда ответные пакеты пропадают и появляются.
4. Сравнить данные с моментами изменения конфигурации МСЭ напарником

Группа 2:

1. Открыть Outpost;
2. Зайти в меню «Параметры» -> «Системные» -> «ICMP параметры» -> отключить/включить эхо-запросы и ответы;
3. Проверить состояние ответов на ping-запросы у напарника;
4. Повторить п.1-2 несколько раз.

Поменяться с напарником ролями и повторить вышеуказанные пункты.

Этап 2. Совместный режим. Каждый студент из группы 1 должен работать в паре со студентом из группы 2.

Группа 1:

1. Запустить утилиту сканирования сети XSpider;
2. В меню «Правка» выбрать «Добавить хост»;
3. Введите IP-адрес хоста напарника;
4. Узнать у напарника текущий режим работы МСЭ;
5. В меню «Сканирование» выберите «Старт все»;

Начнется попытка XSpider сканировать указанный хост. В случае, если на целевом хосте отключены ICMP-ответы, то сканирование происходить не будет без установки в XSpider специальной опции: меню «Профиль» -> «Редактировать текущий» -> «Поиск хостов» -> поставить галочку «Сканировать не отвечающие хосты».

6. Сбросить флаг «Сканировать не отвечающие хосты» для возврата XSpider в первоначальную конфигурацию.

Группа 2:

1. Открыть Outpost;
2. Зайти в меню «Параметры» -> «Системные» -> «ICMP параметры» -> отключить/включить эхо-запросы и ответы;
3. Проверить, как работает XSpider у напарника;
4. Повторить п.1-2 несколько раз для двух режимов XSpider – требующего ICMP-ответа и не требующего.

Поменяться с напарником ролями и повторить вышеуказанные пункты.

По завершению лабораторной работы установленное в процессе занятия ПО необходимо удалить из системы.

3. Отчет

После окончания практической части лабораторной работы студенты должны предоставить письменный отчет, содержащий информацию о проделанной работе и ответы на вопросы в следующем формате:

1. Ф.И.О.
2. Ф.И.О. Напарника
3. Имя и адрес рабочего компьютера.
4. Имя и адрес исследуемого компьютера.
5. Опишите утилиту ping, методы и случаи ее применения.
6. Описать данные, полученные о компьютере напарника с помощью XSpider
7. Какого типа уязвимости были найдены?
8. Как можно предотвратить появление таких уязвимостей с помощью изученных средств?
9. Какие еще сканеры и МСЭ вы знаете? Какие между ними и изученными отличия? (доп +)
10. Выводы.

Контрольные вопросы :

9) Опишите утилиту ping, методы и случаи ее применения.

10)Описать данные, полученные о компьютере с помощью XSpider

11)Опишите типы уязвимостей компьютерных систем

12)Как можно предотвратить появление таких уязвимостей с помощью изученных средств?

13)Описать известные типы МСЭ и отличия между ними.

Практическая работа №4

по учебной дисциплине «Защита информации в системах управления»

Раздел 1. Информационная безопасность информационных систем

Методы и технологии защиты информации открытых информационных систем Виртуальные защищенные сети VPN. Протоколы туннелирования

1. Цель работы

1. Исследовать настройки туннеля VPN GRE по схеме «точка-точка»

2. Подготовка к занятию

1. Изучить (повторить) теоретический материал.
2. Ознакомиться с программой лабораторной работы.
3. Подготовить отчет о лабораторной работе.
4. Ответить на контрольные вопросы.

3. Распределение времени занятия:

Всего: 90 мин

Вступительная часть 2 мин

Проверка готовности студентов к занятию 5 мин

Основная часть – 70 мин

Проверка выполнения практического занятия 10 мин

Заключительная часть 3 мин

4. Правила работы в лаборатории

К работе в лаборатории допускаются лица, изучившие правила и меры безопасности, сдавшие зачет по ним и усвоившие порядок выполнения лабораторной работы.

4.1. Требования безопасности перед началом работ

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории.

- Убедиться в целостности электрических розеток и разъемов. В лаборатории необходимо быть в сменной обуви.

- Включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

4.2. Требования безопасности во время работы

- выполняя практическое занятие, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и носители информации, непосредственно относящиеся к данному лабораторному занятию;

- подключение и отключение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса;

- при обнаружении неисправностей в оборудовании немедленно отключить источники питания и доложить об этом руководителю занятий или лаборанту.

4.3. Требования безопасности по окончании работы

- доложить руководителю занятий или лаборанту о завершении работ;

- привести в порядок и сдать рабочее место лаборанту, и доложить руководству

Примечание для инструктора. Красным или серым цветом выделен текст, который отображается только в копии для инструктора.

Топология

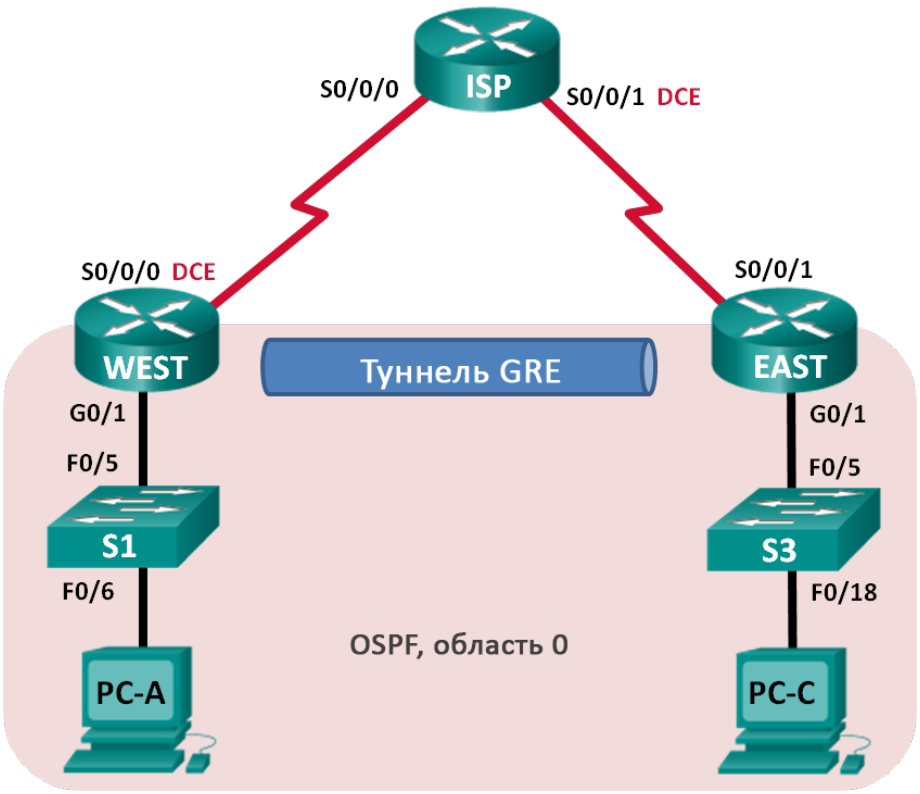


Таблица адресации

Устройство	Интерфейс	IP-адрес	Маска подсети	Шлюз по умолчанию
WEST	G0/1	172.16.1.1	255.255.255.0	Недоступно
	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	Недоступно
	Tunnel0	172.16.12.1	255.255.255.252	Недоступно
ISP	S0/0/0	10.1.1.2	255.255.255.252	Недоступно
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	Недоступно
EAST	G0/1	172.16.2.1	255.255.255.0	Недоступно
	S0/0/1	10.2.2.1	255.255.255.252	Недоступно
	Tunnel0	172.16.12.2	255.255.255.252	Недоступно
PC-A	NIC	172.16.1.3	255.255.255.0	172.16.1.1
PC-C	NIC	172.16.2.3	255.255.255.0	172.16.2.1

Задачи

Часть 1. Базовая настройка устройств

Часть 2. Настройка туннеля GRE

Часть 3. Включение маршрутизации через туннель GRE

Исходные данные/сценарий

Универсальная инкапсуляция при маршрутизации (GRE) — это протокол туннелирования, способный инкапсулировать различные протоколы сетевого уровня между двумя объектами по общедоступной сети, например, в Интернете.

GRE можно использовать с:

подключением сети IPv6 по сетям IPv4

пакетами групповой рассылки, например, OSPF, EIGRP и приложениями потоковой передачи данных

В этой лабораторной работе необходимо настроить незашифрованный туннель GRE VPN «точка-точка» и убедиться, что сетевой трафик использует туннель. Также будет нужно настроить протокол маршрутизации OSPF внутри туннеля GRE VPN. Туннель GRE существует между маршрутизаторами WEST и EAST в области 0 OSPF. Интернет-провайдер не знает о туннеле GRE. Для связи между маршрутизаторами WEST и EAST и интернет-провайдером применяются статические маршруты по умолчанию.

Примечание. В практических лабораторных работах CCNA используются маршрутизаторы с интеграцией сервисов Cisco 1941 (ISR) под управлением ОС Cisco IOS версии 15.2(4) M3 (образ universalk9). В лабораторной работе используются коммутаторы Cisco Catalyst серии 2960 под управлением ОС Cisco IOS 15.0(2) (образ lanbasek9). Допускается использование коммутаторов и маршрутизаторов других моделей, под управлением других версий ОС Cisco IOS. В зависимости от модели устройства и версии Cisco IOS доступные команды и выходные данные могут отличаться от данных, полученных при выполнении лабораторных работ. Точные идентификаторы интерфейсов указаны в сводной таблице интерфейсов маршрутизаторов в конце лабораторной работы.

Примечание. Убедитесь, что предыдущие настройки маршрутизаторов и коммутаторов удалены и они не имеют загрузочных настроек. Если вы не уверены в этом, обратитесь к инструктору.

Примечание для инструктора. Для выполнения инициализации и перезагрузки устройств см. руководство для инструкторов по проведению лабораторных работ.

Необходимые ресурсы:

3 маршрутизатора (Cisco 1941 под управлением ОС Cisco IOS 15.2(4) M3 (образ universal) или аналогичная модель);

2 коммутатора (Cisco 2960 под управлением ОС Cisco IOS 15.0(2), (образ lanbasek9) или аналогичная модель);

2 ПК (под управлением ОС Windows 7, Vista или XP с программой эмуляции терминала, например Tera Term);

консольные кабели для настройки устройств Cisco IOS через порты консоли;

кабели Ethernet и последовательные кабели в соответствии с топологией.

Базовая настройка устройств

В части 1 вам предстоит настроить топологию сети и базовые параметры маршрутизатора, например, IP-адреса интерфейсов, маршрутизацию, доступ к устройствам и пароли.

Подключите кабели в сети в соответствии с топологией.

Выполните инициализацию и перезагрузку маршрутизаторов и коммутаторов.

Произведите базовую настройку маршрутизаторов.

Отключите поиск DNS.

Назначьте имена устройств.

Зашифруйте незашифрованные пароли.

Создайте баннерное сообщение дня (MOTD) для предупреждения пользователей о запрете несанкционированного доступа.

Назначьте **class** в качестве зашифрованного пароля доступа к привилегированному режиму.

Назначьте cisco в качестве пароля для консоли и виртуального терминала VTY и активируйте учётную запись.

Настройте ведение журнала состояния консоли на синхронный режим.

Примените IP-адреса к интерфейсам Serial и Gigabit Ethernet в соответствии с таблицей адресации и активируйте физические интерфейсы. На данном этапе не настраивайте интерфейсы Tunnel0.

Настройте тактовую частоту на **128000** для всех последовательных интерфейсов DCE.

Настройте маршруты по умолчанию к маршрутизатору интернет-провайдера.

WEST(config)# **ip route 0.0.0.0 0.0.0.0 10.1.1.2**

EAST(config)# **ip route 0.0.0.0 0.0.0.0 10.2.2.2**

Настройте компьютеры.

Настройте IP-адреса и шлюзы по умолчанию на всех ПК в соответствии с таблицей адресации.

Проверьте соединение.

На данный момент компьютеры не могут отправлять друг другу эхо-запросы. Каждый ПК должен получать ответ на эхо-запрос от своего шлюза по умолчанию. Маршрутизаторы могут отправлять эхо-запросы на последовательные интерфейсы других маршрутизаторов в топологии. Если это не так, устраните неполадки и убедитесь в наличии связи.

Сохраните текущую конфигурацию.

Контрольные вопросы:

14)Смоделировать и исследовать в PT Visual Studio структуру двухточеченой сети.

15)Сформулировать требования к оборудованию для реализации GRE.

16)Перечислить основные этапы конфигурирования GRE.

17)Синтаксис команд для включения GRE туннеля в Cisco IOS.

18)Реализация защищенного GRE туннеля с использованием IPSec.

19)Настроить статическую маршрутизацию при использовании GRE.

20)Настроить динамическую маршрутизацию при использовании GRE..

Практическая работа №5

по учебной дисциплине «Защита информации в системах управления»

Раздел 1. Информационная безопасность информационных систем

Защита удаленного доступа. Организация защищенных удаленных подключений

1. Цель работы

1. Исследовать принципы настройки базового PPP с аунтификацией.

2. Подготовка к занятию

1. Изучить (повторить) теоретический материал.
2. Ознакомиться с программой лабораторной работы.
3. Подготовить отчет о лабораторной работе.
4. Ответить на контрольные вопросы.

3. Распределение времени занятия:

Всего: 90 мин

Вступительная часть 2 мин

Проверка готовности студентов к занятию 5 мин

Основная часть – 70 мин

Проверка выполнения практического занятия 10 мин

Заключительная часть 3 мин

4. Правила работы в лаборатории

К работе в лаборатории допускаются лица, изучившие правила и меры безопасности, сдавшие зачет по ним и усвоившие порядок выполнения лабораторной работы.

4.1. Требования безопасности перед началом работ

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории.

- Убедиться в целостности электрических розеток и разъемов. В лаборатории необходимо быть в сменной обуви.

- Включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

4.2. Требования безопасности во время работы

- выполняя практическое занятие, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и носители

информации, непосредственно относящиеся к данному лабораторному занятию;

- подключение и отключение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса;

- при обнаружении неисправностей в оборудовании немедленно отключить источники питания и доложить об этом руководителю занятий или лаборанту.

4.3. Требования безопасности по окончании работы

- доложить руководителю занятий или лаборанту о завершении работ;

- привести в порядок и сдать рабочее место лаборанту, и доложить руководству

Настройка базового PPP с аутентификацией.

Топология

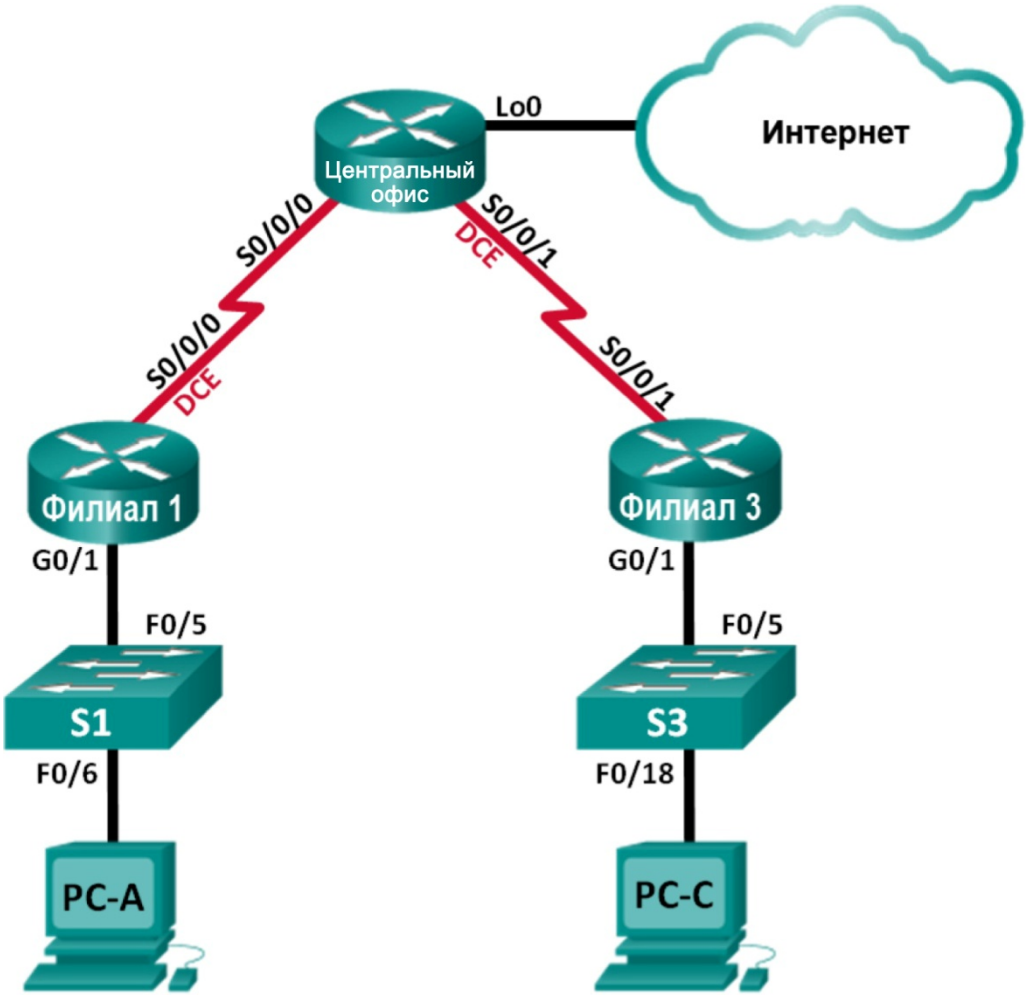


Таблица адресации

Устройство	Интерфейс	IP-адрес	Маска подсети	Шлюз по умолчанию
Филиал 1	G0/1	192.168.1.1	255.255.255.0	Недоступно
	S0/0/0 (DCE)	10.1.1.1	255.255.255.252	Недоступно
Central	S0/0/0	10.1.1.2	255.255.255.252	Недоступно
	S0/0/1 (DCE)	10.2.2.2	255.255.255.252	Недоступно
	Lo0	209.165.200.225	255.255.255.224	Недоступно
Филиал 3	G0/1	192.168.3.1	255.255.255.0	Недоступно
	S0/0/1	10.2.2.1	255.255.255.252	Недоступно
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1
PC-C	NIC	192.168.3.3	255.255.255.0	192.168.3.1

Задачи

Часть 1. Базовая настройка устройств

Часть 2. Настройка инкапсуляции PPP

Часть 3. Настройка аутентификации CHAP PPP

Исходные данные/сценарий

Шаг 1: PPP — очень распространенный протокол WAN уровня 2. PPP можно использовать для подключения из локальной сети к WAN-провайдеру и для подключения сегментов LAN в рамках корпоративной сети.

Шаг 2: В этой лабораторной работе требуется настроить инкапсуляцию PPP на выделенных последовательных каналах между маршрутизаторами филиалов и центральным маршрутизатором. Требуется настроить

протокол аутентификации по квитированию вызова (CHAP) PPP на последовательных каналах PPP. Вы также изучите влияние, оказываемое изменениями инкапсуляции и аутентификации на состояние последовательного канала.

Шаг 3: **Примечание**. В практических лабораторных работах CCNA используются маршрутизаторы с интеграцией сервисов Cisco 1941 (ISR) под управлением ОС Cisco IOS версии 15.2(4) M3 (образ universalk9). В лабораторной работе используются коммутаторы Cisco Catalyst серии 2960 под управлением ОС Cisco IOS 15.0(2) (образ lanbasek9). Допускается использование коммутаторов и маршрутизаторов других моделей, под управлением других версий ОС Cisco IOS. В зависимости от модели устройства и версии Cisco IOS доступные команды и выходные данные могут отличаться от данных, полученных при выполнении лабораторных работ. Точные идентификаторы интерфейсов указаны в сводной таблице интерфейсов маршрутизаторов в конце лабораторной работы.

Шаг 4: **Примечание**. Убедитесь, что предыдущие настройки маршрутизаторов и коммутаторов удалены и они не имеют загрузочных настроек. Если вы не уверены в этом, обратитесь к инструктору.

Шаг 5: **Примечание для инструктора.** Для выполнения инициализации и перезагрузки устройств см. руководство для инструкторов по проведению лабораторных работ.

Необходимые ресурсы:

3 маршрутизатора (Cisco 1941 под управлением ОС Cisco IOS 15.2(4) M3 (образ universal) или аналогичная модель);

2 коммутатора (Cisco 2960 под управлением ОС Cisco IOS 15.0(2), (образ lanbasek9) или аналогичная модель);

2 ПК (под управлением ОС Windows 7, Vista или XP с программой эмуляции терминала, например Tera Term);

консольные кабели для настройки устройств Cisco IOS через порты консоли;

кабели Ethernet и последовательные кабели в соответствии с топологией.

Базовая настройка устройств

Шаг 6: В части 1 вам предстоит настроить топологию сети и базовые параметры маршрутизатора, например, IP-адреса интерфейсов, маршрутизацию, доступ к устройствам и пароли.

Подключите кабели в сети в соответствии с топологией.

Шаг 7: Подключите устройства, как показано в топологии, и подсоедините необходимые кабели.

Выполните инициализацию и перезагрузку маршрутизаторов и коммутаторов.

Произведите базовую настройку маршрутизаторов.

i) Отключите поиск DNS.

ii) Настройте имя устройства.

iii) Зашифруйте незашифрованные пароли.

iv) Создайте баннерное сообщение дня (MOTD) для предупреждения пользователей о запрете несанкционированного доступа.

v) Назначьте **class** в качестве зашифрованного пароля доступа к привилегированному режиму.

vi) Назначьте **cisco** в качестве пароля для консоли и виртуального терминала VTY и активируйте учётную запись.

vii) Настройте ведение журнала состояния консоли на синхронный режим.

viii) Примените IP-адреса к интерфейсам Serial и Gigabit Ethernet в соответствии с таблицей адресации и включите физические интерфейсы.

ix) Настройте тактовую частоту на **128000** для всех последовательных интерфейсов DCE.

x) На маршрутизаторе «Главный» создайте **Loopback 0** для имитации доступа в Интернет и назначьте IP-адрес согласно таблице адресации.

Настройте маршрутизацию.

Практическая работа №6

по учебной дисциплине **«Защита информации в системах управления»**

Раздел 2. Средства реализации защиты в информационных системах

Обнаружение и предотвращение вторжений. Исследование механизмов обнаружения сетевых аномалий

1. Цель работы

1. Исследовать сбор и анализ данных NetFlow

2. Подготовка к занятию

1. Изучить (повторить) теоретический материал.
2. Ознакомиться с программой лабораторной работы.
3. Подготовить отчет о лабораторной работе.
4. Ответить на контрольные вопросы.

3. Распределение времени занятия:

Всего: 90 мин

Вступительная часть 2 мин

Проверка готовности студентов к занятию 5 мин

Основная часть – 70 мин

Проверка выполнения практического занятия 10 мин

Заключительная часть 3 мин

4. Правила работы в лаборатории

К работе в лаборатории допускаются лица, изучившие правила и меры безопасности, сдавшие зачет по ним и усвоившие порядок выполнения лабораторной работы.

4.1. Требования безопасности перед началом работ

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории.

- Убедиться в целостности электрических розеток и разъемов. В лаборатории необходимо быть в сменной обуви.

- Включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

4.2. Требования безопасности во время работы

- выполняя практическое занятие, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и носители информации, непосредственно относящиеся к данному лабораторному занятию;

- подключение и отключение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса;

- при обнаружении неисправностей в оборудовании немедленно отключить источники питания и доложить об этом руководителю занятий или лаборанту.

4.3. Требования безопасности по окончании работы

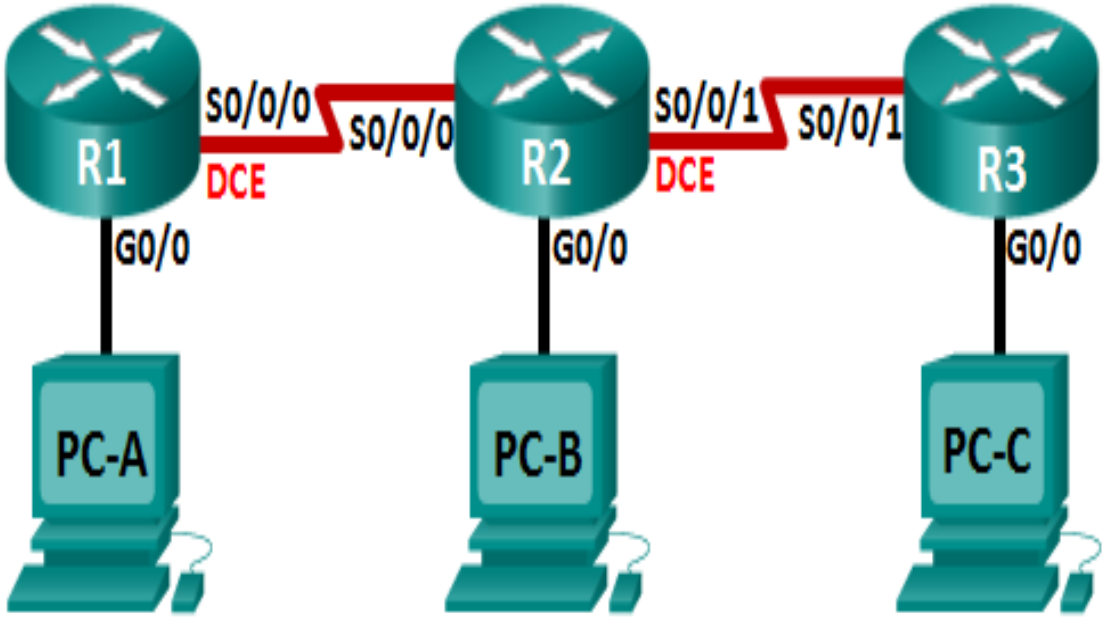
- доложить руководителю занятий или лаборанту о завершении работ;

- привести в порядок и сдать рабочее место лаборанту, и доложить руководству

Сбор и анализ данных NetFlow (вариант для инструктора)

Примечание для инструктора. Красным или серым цветом выделен текст, который отображается только в копии для инструктора.

Топология



Программное обеспечение системы сбора данных и анализатора NetFlow

Таблица адресации

Устройство	Интерфейс	IP-адрес	Шлюз по умолчанию
R1	G0/0	192.168.1.1/24	Недоступно
	S0/0/0 (DCE)	192.168.12.1/30	Недоступно
R2	G0/0	192.168.2.1/24	Недоступно
	S0/0/0	192.168.12.2/30	Недоступно
	S0/0/1 (DCE)	192.168.23.1/30	Недоступно
R3	G0/0	192.168.3.1/24	Недоступно
	S0/0/1	192.168.23.2/30	Недоступно
PC-A	NIC	192.168.1.3	192.168.1.1
PC-B	NIC	192.168.2.3	192.168.2.1
PC-C	NIC	192.168.3.3	192.168.3.1

Задачи

Часть 1. Создание сети и настройка базовых параметров устройств

Часть 2. Настройка NetFlow на маршрутизаторе

Часть 3. Анализ NetFlow с помощью интерфейса командной строки

Часть 4. Изучение ПО сбора данных и анализатора NetFlow

Исходные данные/сценарий

NetFlow — это технология Cisco IOS, предоставляющая статистические данные о пакетах, проходящих через маршрутизатор или многоуровневый коммутатор Cisco. NetFlow обеспечивает контроль сети и безопасности, планирование сетевых ресурсов, анализ трафика и учёт IP. Важно не путать назначение и результаты NetFlow с назначением и результатами оборудования и программного обеспечения для сбора пакетов. Средства сбора пакетов записывают всю входящую и исходящую информацию

сетевого устройства для последующего анализа, в то время как NetFlow собирает только определённую статистическую информацию.

Flexible NetFlow — это новейшая версия технологии NetFlow, которая расширяет возможности первоначального протокола NetFlow, позволяя настраивать параметры анализа трафика. Flexible NetFlow использует формат экспорта версии 9. Начиная с Cisco IOS версии 15.1, поддерживаются многие полезные команды Flexible NetFlow.

В этой лабораторной работе вам потребуется настроить NetFlow для сбора данных входящих и исходящих пакетов. С помощью команды **show** вы сможете проверить, что NetFlow находится в рабочем состоянии и осуществляет сбор статистических данных. Вы также рассмотрите доступные варианты ПО сборщика данных и анализатора NetFlow.

Примечание. В практических лабораторных работах CCNA используются маршрутизаторы с интеграцией сервисов Cisco 1941 (ISR) под управлением ОС Cisco IOS версии 15.2(4) M3 (образ universalk9). Возможно использование других маршрутизаторов и версий Cisco IOS. В зависимости от модели устройства и версии Cisco IOS доступные команды и выходные данные могут отличаться от данных, полученных при выполнении лабораторных работ. Точные идентификаторы интерфейсов указаны в сводной таблице интерфейсов маршрутизаторов в конце лабораторной работы.

Примечание. Убедитесь, что предыдущие настройки маршрутизаторов и коммутаторов удалены, и они не содержат файла загрузочной конфигурации. Если вы не уверены в этом, обратитесь к инструктору.

Контрольные вопросы:

21)Этапы и средства реализации атак. Классификация атак.

22)Таксономия систем обнаружения атак.

23)Совместное применение систем обнаружения атак и других средств защиты.

24)Методы обнаружения аномалий: статистический анализ, нейросетевые методы, анализ изменения критических параметров во времени.

25)Анализ журналов регистрации и сетевого трафика.

26)Анализ заголовков, процессов, сервисов и портов.

27)Настроить NetFlow на маршрутизаторе.

28)Перечислить семь основных полей, используемых первоначальным протоколом NetFlow для различения потоков данных.

Практическая работа №7
по учебной дисциплине **«Защита информации в системах управления»**

Раздел 2. Средства реализации защиты в информационных системах

Защита от вредоносных программ и спама

Цель работы: Исследование механизмов работы антивирусных программ

1. Подготовка к занятию

1. Изучить (повторить) теоретический материал.
2. Ознакомиться с программой лабораторной работы.
3. Подготовить отчет о лабораторной работе.
4. Ответить на контрольные вопросы.

2. Распределение времени занятия:

Всего: 90 мин

Вступительная часть 2 мин

Проверка готовности студентов к занятию 5 мин

Основная часть – 70 мин

Проверка выполнения практического занятия 10 мин

Заключительная часть 3 мин

4. Правила работы в лаборатории

К работе в лаборатории допускаются лица, изучившие правила и меры безопасности, сдавшие зачет по ним и усвоившие порядок выполнения лабораторной работы.

4.1. Требования безопасности перед началом работ

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории.

- Убедиться в целостности электрических розеток и разъемов. В лаборатории необходимо быть в сменной обуви.

- Включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

4.2. Требования безопасности во время работы

- выполняя практическое занятие, студенты обязаны использовать только вычислительную технику, периферийное оборудование,

соединительные кабели, измерительное оборудование и носители информации, непосредственно относящиеся к данному лабораторному занятию;

- подключение и отключение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса;

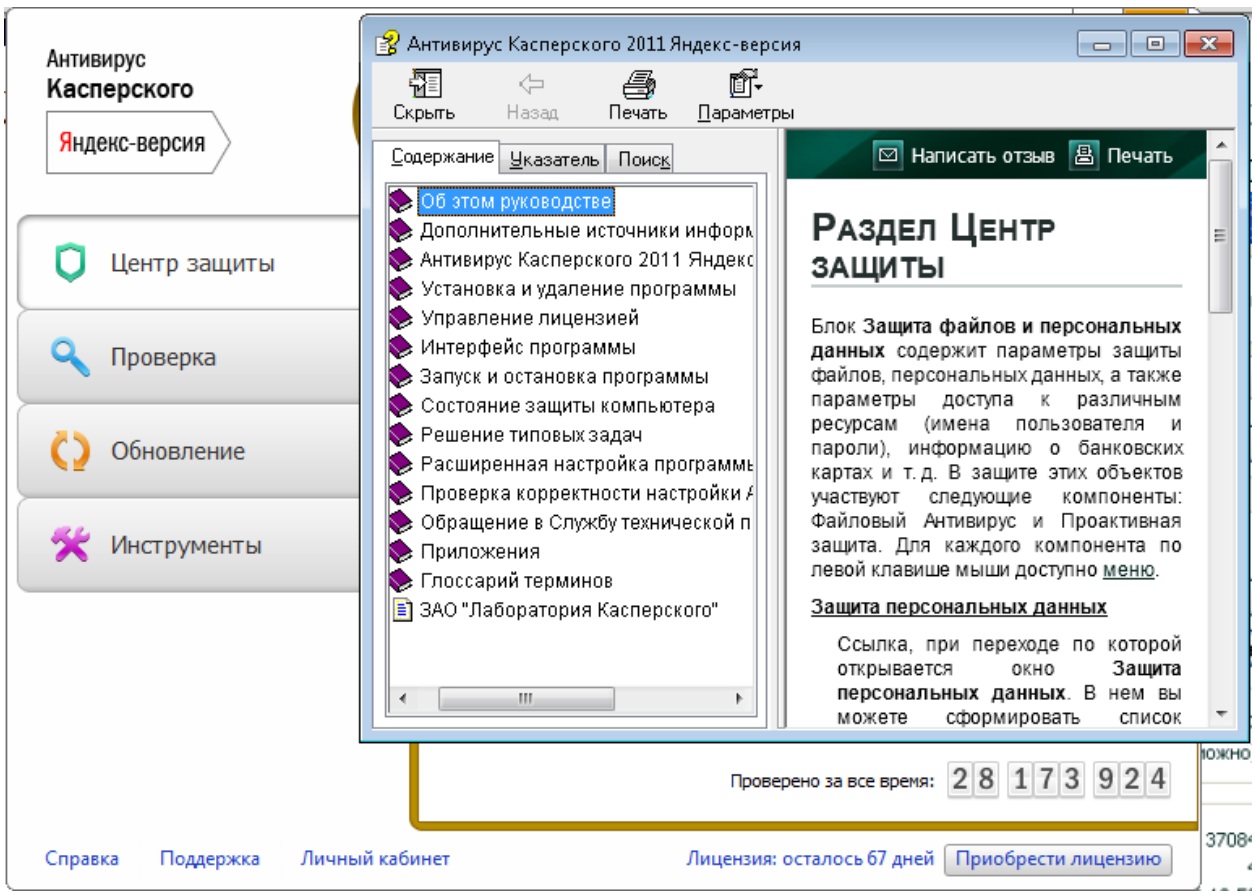
- при обнаружении неисправностей в оборудовании немедленно отключить источники питания и доложить об этом руководителю занятий или лаборанту.

4.3. Требования безопасности по окончании работы

- доложить руководителю занятий или лаборанту о завершении работ;

- привести в порядок и сдать рабочее место лаборанту, и доложить руководству

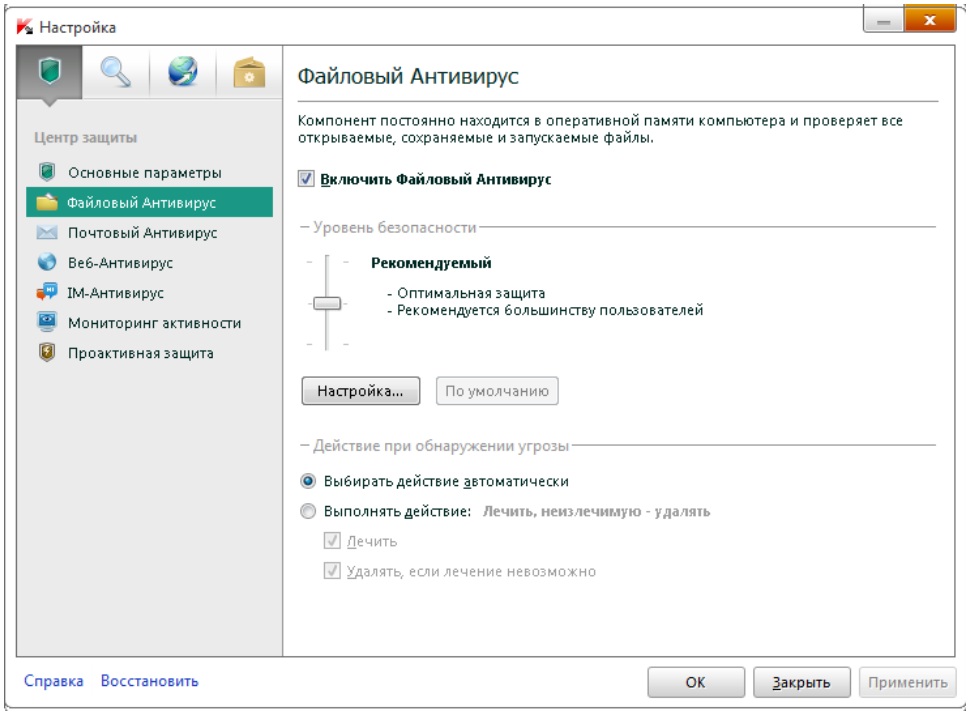
1. Запускаем антивирусную программу Антивирус Касперского 2011 Яндекс – версия (Пуск - Антивирус Касперского 2011).



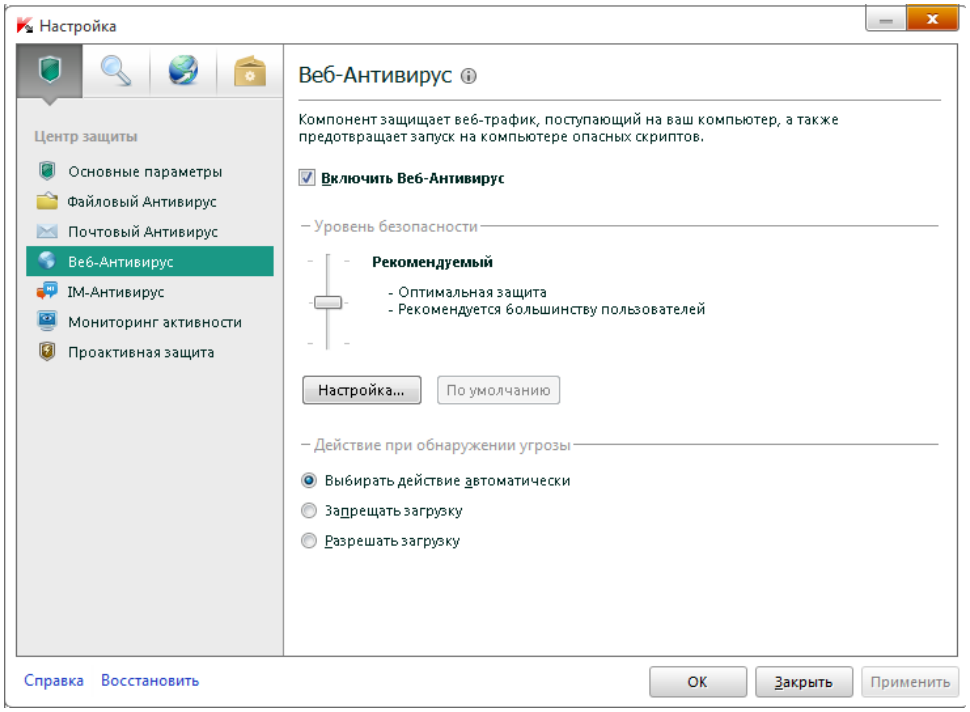
2. Используя Меню Справка (в левом нижнем углу) изучаем команды программы.

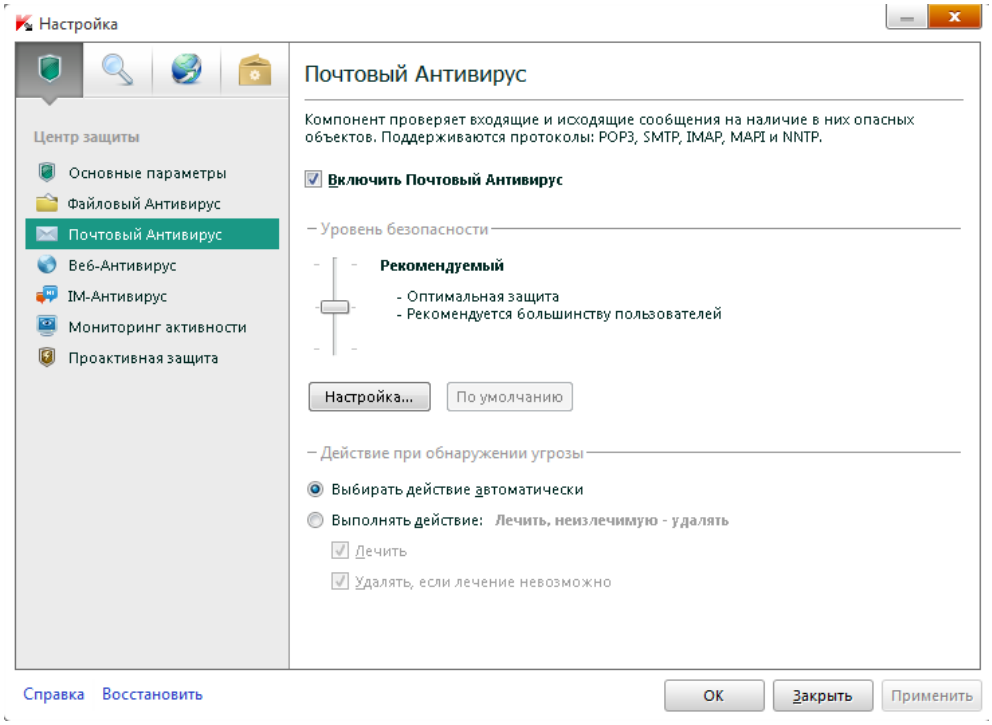
К основными функциями программы относятся: включение и отключение компонентов защиты, выполнение задач проверки на вирусы, обновление баз и модулей программы и т. д.

3. Настройка защиты файлов и персональных данных. На главном окне программы выбираем вкладку Центр защиты, заходим в пункт Защита файлов и персональных данных – Файловый антивирус – Настроить. Отмечаем флажком пункт Включить Файловый Антивирус и устанавливаем необходимый уровень безопасности. Нажимаем Enter.

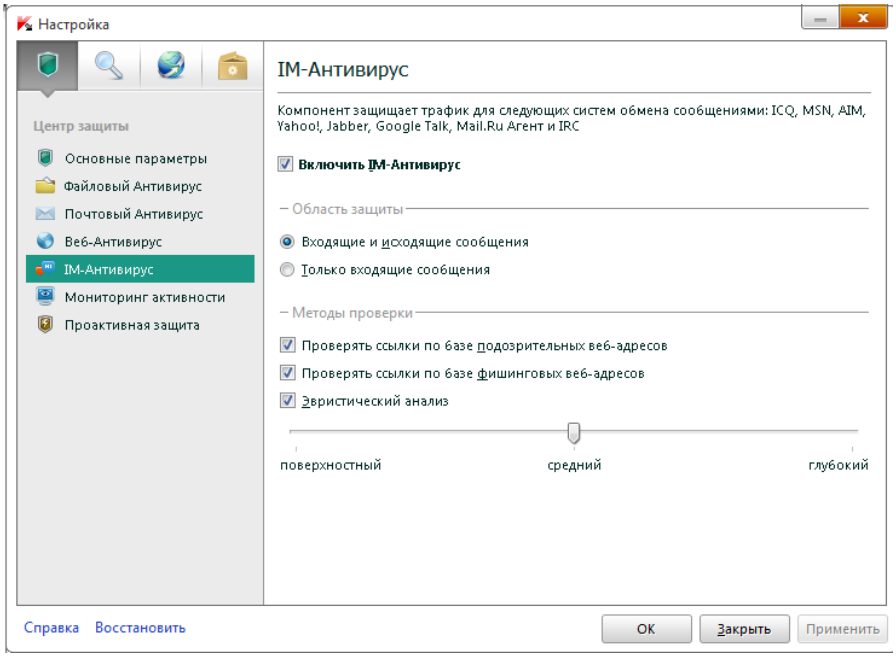


4. Аналогичным образом устанавливаем параметры защиты для систем и программ (пункты Веб-антивирус и Почтовый антивирус).

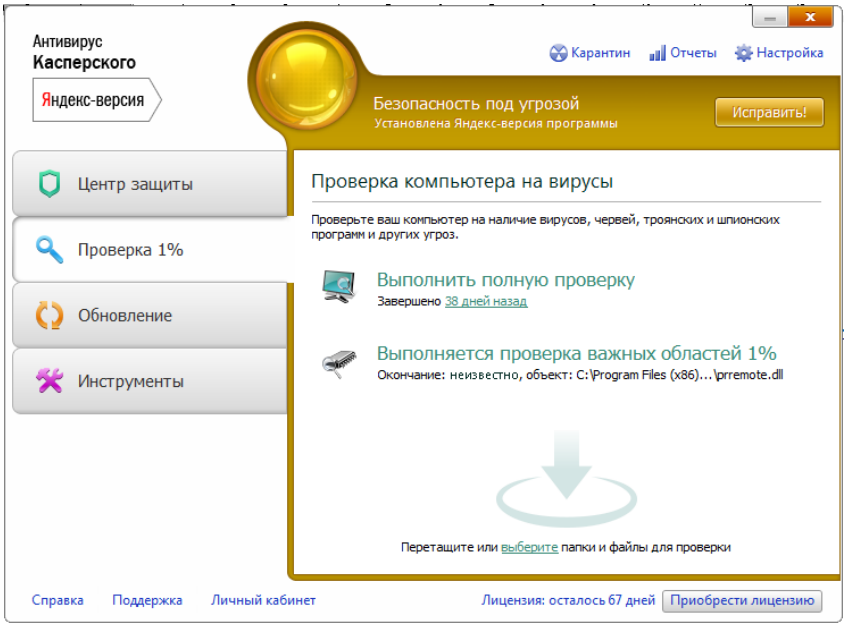




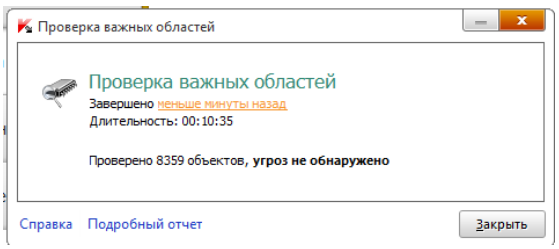
5. Вкладка Контроль работы в сети позволяет настроить программу для безопасного просмотра веб-сайтов, онлайн общения, использования программ электронной почты и платежных систем. Выполним настройку IM-антивируса:



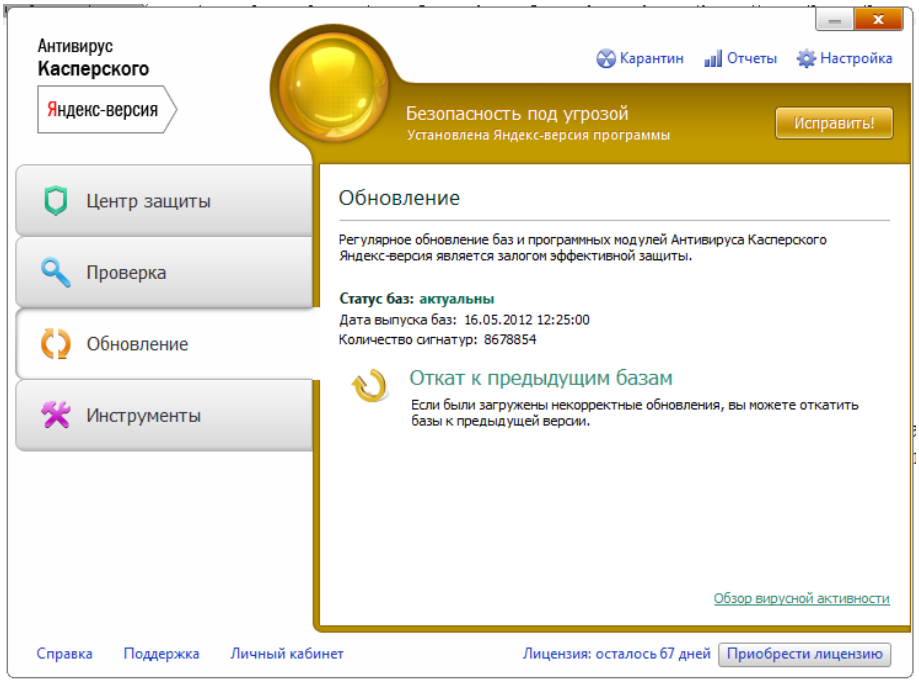
6. Проверка на вирусы. На вкладке Проверка выбираем пункт Выполнить проверку важных областей.



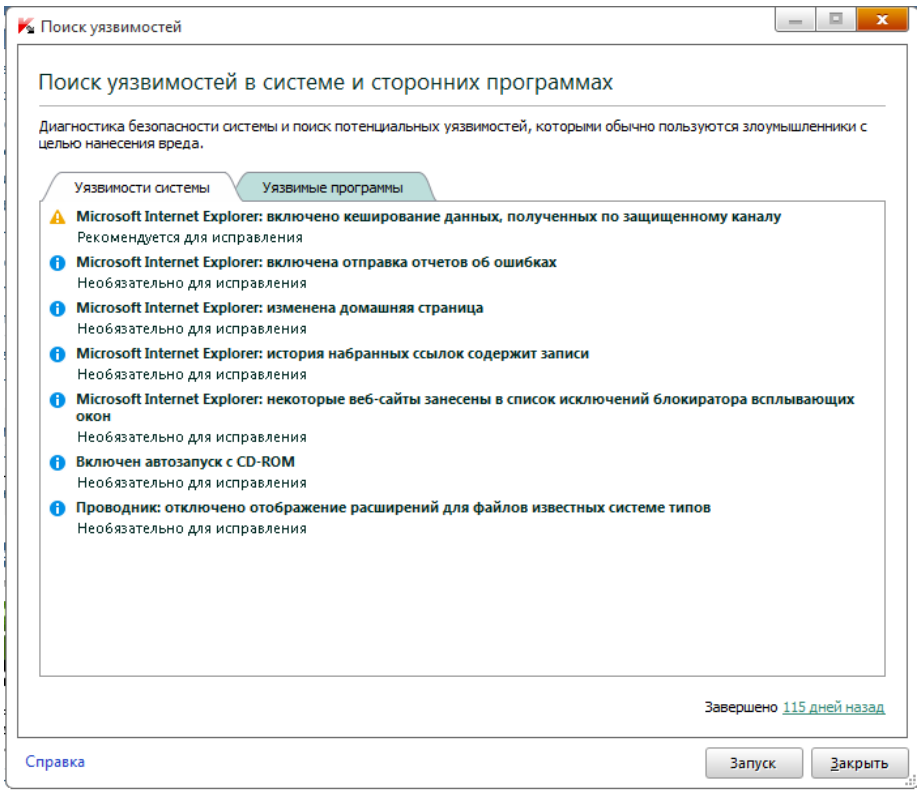
Результат:



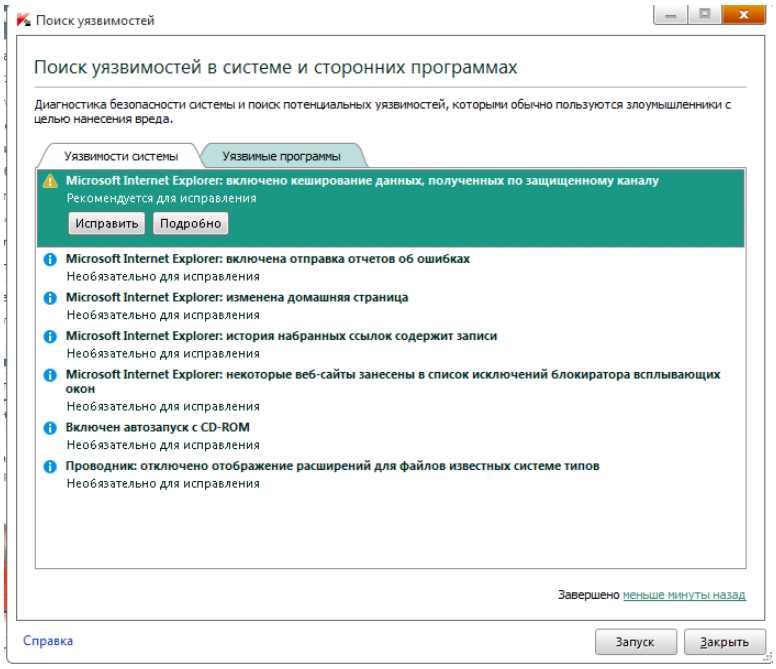
7. Обновление баз и модулей программы. На вкладке Обновление указан статус загруженных баз и программных модулей. Обновление баз в данной программе происходит автоматически при подключении к сети.



8. Поиск уязвимостей в системе. На вкладке Инструменты представлены инструменты и сервисы предоставляющие дополнительные возможности для обеспечения безопасности компьютера. Среди них Создание диска аварийного восстановления, Поиск уязвимостей в системе, Настройка браузера, Устранение следов активности и Восстановление после заражения. Воспользуемся сервисом Поиск уязвимостей.


Поиск уязвимостей
Поиск уязвимостей в системе и сторонних программах
Диагностика безопасности системы и поиск потенциальных уязвимостей, которыми обычно пользуются злоумышленники с целью нанесения вреда.
Уязвимости системы
Уязвимые программы
Microsoft Internet Explorer: включено кеширование данных, полученных по защищенному каналу
Рекомендуется для исправления
Microsoft Internet Explorer: включена отправка отчетов об ошибках
Необязательно для исправления
Microsoft Internet Explorer: изменена домашняя страница
Необязательно для исправления
Microsoft Internet Explorer: история набранных ссылок содержит записи
Необязательно для исправления
Microsoft Internet Explorer: некоторые веб-сайты занесены в список исключений блокиратора всплывающих окон
Необязательно для исправления
Включен автозапуск с CD-ROM
Необязательно для исправления
Проводник: отключено отображение расширений для файлов известных системе типов
Необязательно для исправления
Завершено 115 дней назад
Справка
Запуск
Закрыть

Результат:


Поиск уязвимостей
Поиск уязвимостей в системе и сторонних программах
Диагностика безопасности системы и поиск потенциальных уязвимостей, которыми обычно пользуются злоумышленники с целью нанесения вреда.
Уязвимости системы
Уязвимые программы
Microsoft Internet Explorer: включено кеширование данных, полученных по защищенному каналу
Рекомендуется для исправления
Исправить
Подробно
Microsoft Internet Explorer: включена отправка отчетов об ошибках
Необязательно для исправления
Microsoft Internet Explorer: изменена домашняя страница
Необязательно для исправления
Microsoft Internet Explorer: история набранных ссылок содержит записи
Необязательно для исправления
Microsoft Internet Explorer: некоторые веб-сайты занесены в список исключений блокиратора всплывающих окон
Необязательно для исправления
Включен автозапуск с CD-ROM
Необязательно для исправления
Проводник: отключено отображение расширений для файлов известных системе типов
Необязательно для исправления
Завершено меньше минуты назад
Справка
Запуск
Закрыть

Контрольные вопросы:

1) Сформулировать алгоритм сигнатурного поиска вредоносного ПО.
2) Сформулировать алгоритм эвристического поиска вредоносного ПО.
3) Сформулировать алгоритм работы веб-сканера антивируса
4) Сформулировать алгоритм работы почтового сканера антивируса.
5) Разработать и осуществить эмпирический анализ алгоритма сортировки простыми вставками.
6) Разработать и осуществить эмпирический анализ алгоритма бинарной сортировки.
7) Разработать алгоритм быстрой сортировки двумерного массива и осуществить математический анализ.
8) Разработать алгоритм пирамидальной сортировки двумерного массива и осуществить математический анализ.

Практическая работа №8
по учебной дисциплине **«Защита информации в системах управления»**

Раздел 2. Средства реализации защиты в информационных системах

Защита информации в СУБД. Настройка безопасности в MySQL

1.Цель работы: Ознакомиться с приложениями, включенными в состав СУБД MySQL. Получить навыки управления учетными записями пользователей и определения привилегий. Ознакомиться с утилитами, входящими в состав СУБД MySQL, получить навыки работы с ними.

2. Подготовка к занятию

1. Изучить (повторить) теоретический материал.
2. Ознакомиться с программой лабораторной работы.
3. Подготовить отчет о лабораторной работе.
4. Ответить на контрольные вопросы.

3. Распределение времени занятия:

Всего: 90 мин

Вступительная часть 2 мин

Проверка готовности студентов к занятию 5 мин

Основная часть – 70 мин

Проверка выполнения практического занятия 10 мин

Заключительная часть 3 мин

4. Правила работы в лаборатории

К работе в лаборатории допускаются лица, изучившие правила и меры безопасности, сдавшие зачет по ним и усвоившие порядок выполнения лабораторной работы.

4.1. Требования безопасности перед началом работ

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории.
- Убедиться в целостности электрических розеток и разъемов. В лаборатории необходимо быть в сменной обуви.

- Включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

4.2. Требования безопасности во время работы

- выполняя практическое занятие, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и носители информации, непосредственно относящиеся к данному лабораторному занятию;
- подключение и отключение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса;
- при обнаружении неисправностей в оборудовании немедленно отключить источники питания и доложить об этом руководителю занятий или лаборанту.

4.3. Требования безопасности по окончании работы

- доложить руководителю занятий или лаборанту о завершении работ;
- привести в порядок и сдать рабочее место лаборанту, и доложить руководству

Запуск MySQL

Управление сервером обычно осуществляется из командной строки. Запуск в Windows осуществляется через сеанс командной строки выполнением следующей команды:

```
D:\usr\local\Mysql\bin\mysqld --standalone
```

Эта команда запустит демон mysql в фоновом режиме. В Windows 95/98 не предусмотрен запуск mysqld в виде службы. В Windows 2000 демон mysql запускается в виде службы.

Можно осуществить запуск winmysqladmin.exe, в этом случае все настройки перечисляются в файле my.ini

При запуске mysqld можно указывать следующие опции:

Таблица 1- Опции команды MySQLD

`-?, --help`	Справка
`-b, --basedir=[path]`	Путь к каталогу в котором установлен mysql
`-h, --datadir [homedir]`	Путь к каталогу, в котором хранятся базы данных.
`-l, --log=[filename]`	Имя журнала транзакций
`-L, --language=[language]`	Язык по умолчанию(обычно English).
`-P, --port=[port]`	Порт для соединения.
`--skip-grant-tables`	Игнорировать таблицы привилегий. Это дает

	любому ПОЛНЫЙ доступ ко всем таблицам. Не следует предоставлять обычным пользователям разрешений на запуск mysqld.
`--skip-name-resolve`	Позволяет предоставлять доступ только тем хостам, чьи IP-адреса указаны в таблицах привилегий. Ипользуется для более высокого уровня защиты.
`--skip-networking`	Использовать подключения только через интерфейс localhost.
`-V, --version`	Вывести информацию о версии.

Наличие в статусной строке иконки светофора с активным зеленым цветом указывает на то, что сервер запущен (см. рис 1).

Рисунок 1 - Приложение winmysqladmin запущено

Теперь можно попытаться войти в сервер. В случае, если предполагается управление сервером через консоль, то необходимо использовать команду **mysql**. Изначально существует единственный пользователь, которому предоставляется право входа - **root**, которая не имеет пароля. Первое, что нужно сделать войти под именем **root** и зарегистрировать нового пользователя и установить для него пароль. Команда **mysql** может использовать следующие опции:

Таблица 2 - Опции команды MySQL

-?, --help	Справка
-h,--hostname=[hostname]	Имя сервера mysql.
-u, --user=[user]	Имя пользователя для доступа к mysql.
-p, --password=[password]	Пароль пользователя для доступа к mysql.
-P, --port=[port]	Порт для соединения с сервером.
-V, --version	Информация о версии

Примечание. Команды mysqld и mysql имеют еще некоторые опции, но в данный момент они особого интереса не представляют.

Запуск из сеанса ДОС осуществляется как показано на Рисунок 2 (в указанном случае осуществляется подключение к БД mysql).

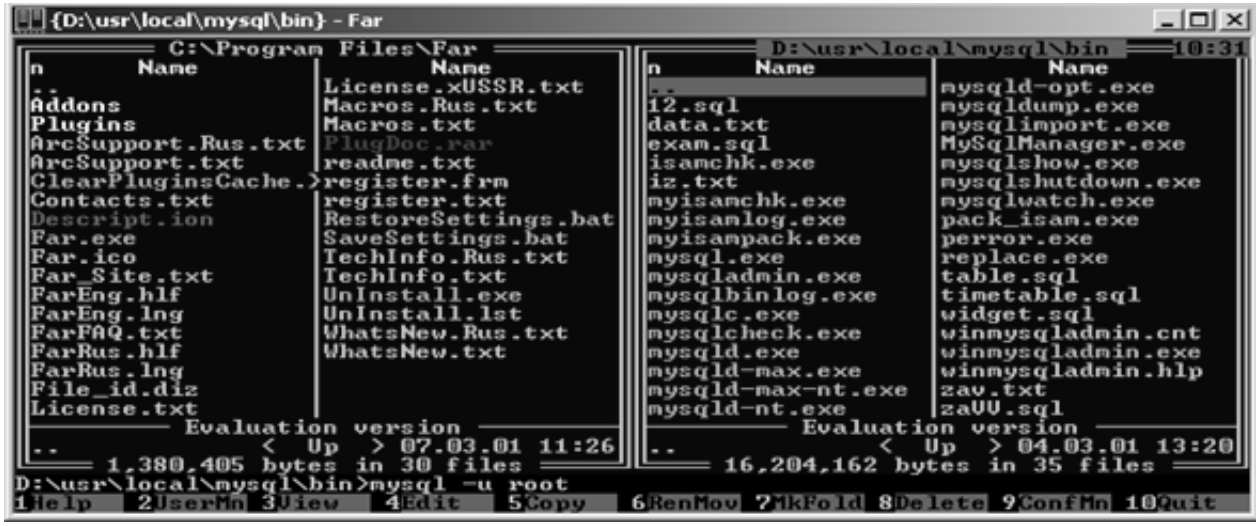


Рисунок 2 - Запуск консоли MYSQL

Для выполнения в строке наберите команду: **mysql –u root**

Рисунок 3 - Успешный запуск консоли

Если вы это получили, значит вы успешно вошли в консоль mysql, которая используется для администрирования сервера.

Для составления отчета вам понадобятся приведение команд, которые вы будете посылать на сервер. В MySQL имеется возможность ведение протокола выполняемых команд, чтобы запустить ведение протокола необходимо выполнить команду

\T filename

!!! обязательно в верхнем регистре. Filename – имя файла, в который будут записываться команды (создается автоматически при выполнении команды, и действует во время жизни сеанса, т.е. в случае отключения от сервера лог прерывается и для возобновления необходимо повторить команду с выводом в новый файл, так как команда затирает имеющиеся в файле данные).

Просмотр списка БД, доступных на сервере осуществляется командой ***SHOW DATABASES.***

Для выполнения в строке наберите команду: **show databases.**

Командой: **USE MYSQL;** – выбираем текущую БД где MYSQL имя БД.

Система привилегий и безопасность в MySQL

- User
- Db
- Host
- Пользовательские привилегии

База данных mysql и таблицы привилегий.

Итак, вы успешно вошли в базу данных mysql, которая используется для администрирования сервера. Что же здесь находится? А находятся здесь 5 таблиц, которые ничем не отличаются от других таблиц баз данных, за исключением того, что эти таблицы используются для предоставления доступа к базам данных и таблицам в них пользователям. Рассмотрим каждую из них.

Введите следующую команду, **show tables,** которая покажет таблицы в базе данных mysql.

Кратко рассмотрим функции каждой из таблиц:

Таблица User

Определяет, разрешено ли пользователю, пытающемуся подключиться к серверу делать это. Содержит имя пользователя, пароль а также привилегии. Если ввести команду show columns from user; то получим следующее:

Таблица 3- Структура таблицы User

Field	Type	Null	Key	Default	Extra
Host	char(60)		PRI		
User	char(16)		PRI		
Password	char(41)				
Select_priv	enum('N','Y')			N	
Insert_priv	enum('N','Y')			N	
Update_priv	enum('N','Y')			N	
Delete_priv	enum('N','Y')			N	
Create_priv	enum('N','Y')			N	
Drop_priv	enum('N','Y')			N	
Reload_priv	enum('N','Y')			N	
Shutdown_priv	enum('N','Y')			N	
Process_priv	enum('N','Y')			N	
File_priv	enum('N','Y')			N	
Grant_priv	enum('N','Y')			N	
References_priv	enum('N','Y')			N	
Index_priv	enum('N','Y')			N	
Alter_priv	enum('N','Y')			N	
Show_db_priv	enum('N','Y')			N	
Super_priv	enum('N','Y')			N	
Create_tmp_table_priv	enum('N','Y')			N	
Lock_tables_priv	enum('N','Y')			N	
Execute_priv	enum('N','Y')			N	
Repl_slave_priv	enum('N','Y')			N	

Repl_client_priv	enum('N','Y')			N	
Create_view_priv	enum('N','Y')			N	
Show_view_priv	enum('N','Y')			N	
Create_routine_priv	enum('N','Y')			N	
Alter_routine_priv	enum('N','Y')			N	
Create_user_priv	enum('N','Y')			N	
Event_priv	enum('N','Y')			N	
Trigger_priv	enum('N','Y')			N	
ssl_type	enum('','ANY','X509','SPECIFIED')				
ssl_cipher	blob			NULL	
x509_issuer	blob			NULL	
x509_subject	blob			NULL	
max_questions	int(11) unsigned			0	
max_updates	int(11) unsigned			0	
max_connections	int(11) unsigned			0	
max_user_connections	int(11) unsigned			0	

Изначально эта таблица содержит пользователя root без пароля. По умолчанию root может входить с любого хоста, имеет все привилегии и доступ ко всем базам данных.Также в таблице содержится запись для пользователя '%'.

В БД MYSQL содержатся таблицы, называемых таблицами привилегий. Система привилегий будет подробно рассмотрена в следующих работах, а пока вы можете выполнить команды на добавления своего пользователя: Для добавления нового пользователя **your_name**, можно выполнить следующие операторы языка (Insert):

Insert into user (host, user, password, ssl_cipher, x509_issuer, x509_subject)

values ('localhost', 'your_name', password('your_pass'), '', '', '');

Выполнением команды

Select host, user, password from user;

Мы выводим перечисленные поля в виде таблицы

Host	User	Password
%	root	456g879k34df9

Если необходимо выделить все столбцы таблицы, то необходимо набрать * в качестве аргумента команды ***select***.

Чтобы изменения вступили в силу нужно перегрузить сервер, предварительно закончив текущий сеанс работы командой ***quit***.

mysqladmin -u root reload (эта команда перегружает сервер)

После установки пароля для пользователя нужно перезагрузить сервер командой mysqladmin reload, чтобы изменения вступили в силу. После этого можно попробовать войти снова:

Mysql/bin/***mysql -u your_name -p mysql***
Enter password:*******

Если же после этой операции вы не получите приглашение ко входу, то необходимо будет повторить вход в сервер под учетной записью **ROOT** и назначить необходимые права**.** Т.о., недостаточно добавить сведения о пользователе в системную БД, дополнительно необходимо назначить права пользователю, после чего можно начинать настраивать таблицы привилегий, вводить новых пользователей, создавать базы данных и таблицы, то есть делать все то, что называется администрированием. Назначить права можно указанием инструкцией ***INSERT*** для заполнения соответствующие привилегии (перечень привилегий см.

Таблица 3)

Mysql/bin/***mysql -u root***

И выполнить следующий запрос к БД:

Mysql>***USE MYSQL;***

Mysql>***GRANT ALL PRIVILEGES ON *.* TO 'your_name'@'localhost' IDENTIFIED BY 'your_pass' WITH GRANT OPTION;***

Mysql>***FLUSH PRIVILEGES;***

Если пароль был случайно забыт, чтобы его задать по новой, придется стереть файлы mysql.frm mysql.MYI и mysql.MYD из папки с базами данных, затем запустить скрипт mysql_install_db и повторить все по новой. Можно воспользоваться ключом MYSQL и ввести **--skip-grant-tables,** при этом все пароли будут имеет пустое поле.

Команда имеет вид ***mysqld --skip-grant-tables.***

Пояснения:

1.Команда insert вставляет данные в таблицу, не забывайте завершать команды ';'.

2.При вводе пароля используйте функцию password(), иначе пароль работать не будет!

3.Все пароли шифруются mysql, поэтому в поле Password вы видите абракадабры. Это делается в целях безопасности.

4.Не есть хорошей практикой назначать привилегии пользователям в таблице user, так как в этом случае они являются глобальными и распространяются на все базы данных. Предоставляйте привилегии каждому пользователю к конкретной базе данных в таблице db, которая будет рассмотрена далее.

5.При задании имени хоста для входа через сеть рекомендуется явно указывать полное имя хоста, а не '%'. В приведенном выше примере пользователю mary разрешается вход на сервер со всех машин домена

tomsk.ru. Можно также указывать IP-адреса машин и маски подсетей для большей безопасности.

Таблица Db

Определяет к каким базам данных каким пользователям и с каких хостов разрешен доступ. В этой таблице можно предоставлять каждому пользователю доступ к базам данных и назначать привилегии. Если выполнить команду ***show columns from db;*** получим следующее:

Таблица 4 - Структура таблицы Db

Field	Type	Null	Key	Default	Extra
Host	char(60)		PRI		
Db	char(32)		PRI		
User	char(16)		PRI		
Select_priv	char(1)			N	
Insert_priv	char(1)			N	
Update_priv	char(1)			N	
Delete_priv	char(1)			N	
Create_priv	char(1)			N	
Drop_priv	char(1)			N	

— По умолчанию, все привилегии установлены в 'N'. Например, предоставим юзеру mary доступ к базе данных mysql и дадим ему привилегии **select**, **insert** и **update** (описание основных команд mysql будет дано в следующих лабораторных работах, сейчас ваша цель увидеть, как работают таблицы привилегий).

— Для справки:

```
Insert into db (host, user, db, select_priv, insert_priv, update_priv)
Values (''localhost', 'your_name', mysql, 'Y', 'Y', 'Y');
```

— Привилегии, устанавливаемые в таблице db, распространяются только на базу данных library. Если же установить эти привилегии в таблице

user, то они будут распространяться и на другие базы данных, даже если доступ к ним и не установлен явно.

Таблица Host

Таблица host используется для расширения диапазона доступа в таблице db. К примеру, если доступ к какой-либо базе данных должен быть предоставлен более чем одному хосту, тогда следует оставить пустой колонку host в таблице db, и внести в таблицу host необходимые имена хостов. Выполним команду

show columns from host;

Таблица 5 - Структура таблиц Host

Field	Type	Null	Key	Default	Extra
Host	char(60)		PRI		
Db	char(32)		PRI		
Select_priv	char(1)			N	
Insert_priv	char(1)			N	
Update_priv	char(1)			N	
Delete_priv	char(1)			N	
Create_priv	char(1)			N	
Drop_priv	char(1)			N	

Как видно из таблицы, здесь также можно задавать привилегии для доступа к базе данных.Они обычно редко используются без необходимости. Все привилегии доступа нужно задавать в таблице db для каждого пользователя, а в таблице host только перечислить имена хостов.Сервер читает все таблицы, проверяет имя пользователя, пароль, имя хоста, имя базы данных, привилегии.Если в таблице db привилегии select, insert установлены в 'Y', а в таблице host в 'N', то в итоге юзер все равно получит 'Y'.Чтобы не вносить путаницы, лучше назначать привилегии в таблице db.

Эти 3 таблицы являются основными.В новых версиях MySQL, начиная с 3.22 добавлены еще 2 таблицы- tables_priv и columns_priv, которые позволяют задать права доступа к определенной таблице в базе данных и даже к

определенной колонке. Они работают подобно таблице db, только ссылаются на таблицы и колонки. Также, начиная с версии 3.22 можно использовать команду GRANT для предоставления доступа к базам данных, таблицам и колонкам таблиц, что избавляет от необходимости вручную модифицировать таблицы db, tables_priv и columns_priv. Команда GRANT будет подробно рассмотрена в следующих разделах.

Привилегии, предоставляемые MySQL

Таблица 6 - Привилегии пользователя

Привилегия	**Колонка**	**Где используется**
select	Select_priv	таблицы
insert	Insert_priv	таблицы
Update	Update_priv	таблицы
delete	Delete_priv	таблицы
index	Index_priv	таблицы
alter	Alter_priv	таблицы
create	Create_priv	БД, таблицы, индексы
drop	Drop_priv	БД или таблицы
grant	Grant_priv	БД или таблицы
References	References_priv	БД или таблицы
reload	Reload_priv	администрирование сервера
Shutdown	Shutdown_priv	администрирование сервера
Process	Process_priv	администрирование сервера
file	File_priv	доступ к файлам на сервере

Основные утилиты MySQL.

В состав дистрибутива MySQL входят следующие утилиты:

— mysqld
— mysql
— mysqladmin
— mysqlaccess
— mysqlshow
— mysqldump
— isamchk

Утилиты **mysqld** и **mysql** были подробно рассмотрены ранее, поэтому возвращаться к ним не будем. Кратко рассмотрим остальные.

Mysqladmin

Утилита для администрирования сервера. Может использоваться администратором, а также некоторыми пользователями, которым предоставлены определенные привилегии, например – **Reload_priv, Shutdown_priv, Process_priv** и **File_priv**. Данная команда может использоваться для создания баз данных, изменения пароля пользователя(администратор может изменить пароль любому пользователю, а рядовой пользователь – только свой собственный), перезагрузки и остановки сервера, просмотра списка процессов, запущенных на сервере. Mysqladmin поддерживает следующие команды:

Таблица 7 - Опции команды MySQLadmin

Create [database_name]	Создает базу данных
Drop [database_name]	Удаляет базу данных и все таблицы в ней
Reload	Перезагружает сервер
Shutdown	Останавливает работу сервера MySQL
Processlist	Выводит список процессов на сервере
Status	Выводит сообщение о статусе сервера

Пример использования mysqladmin для изменения пароля:

mysqladmin -u your_name password your_pass

Следует заметить, что в случае использования mysqladmin для установки пароля, не требуется использование функции password().Mysqladmin сам заботится о шифровании пароля.

Mysqlaccess

Используется для проверки привилегий пользователя для доступа к конкретной базе данных. Общий синтаксис:

mysqlaccess [host] [user] [db] опции

Полезная утилита для проверки прав доступа пользователя, если он получает сообщение Access denied, при попытке соединиться с базой данных.

Опции:

Таблица 8 - Опции команды MySQLAccess

-?, --help	Справка
-u, --user=[username]	Имя пользователя
-p, --password=[password]	Пароль пользователя
-h, --host=[hostname]	Имя хоста для проверки прав доступа
-d, --db=[dbname]	Имя базы данных для проверки прав доступа
-U, --superuser=[susername]	Имя суперпользователя(root)
-P, --spassword=[spassword]	Пароль администратора
-b, --brief	Выводит краткие сведения о таблице

Mysqlshow

Используется, чтобы показать, с какими базами данных работает сервер, какие таблицы содержит каждая БД и какие колонки есть в каждой таблице.Синтаксис:

mysqlshow [опции] [database [table [field]]]

Mysqlshow может использовать следующие параметры:

Таблица 9 - Параметры команды Mysqlshow

-?, --help	Справка
-h, --host=[hostname]	Имя сервера
-k, --key	Показать ключи для таблицы
-p, --password=[password]	Пароль пользователя
-u, --user=[username]	Имя пользователя
-p, --port=[port]	Порт для связи
-V, --version	Вывести информацию о версии

Если ввести mysqlshow без аргументов, будут показаны все базы данных, если указать имя БД, будут показаны все таблицы в ней.

Команды

mysqlshow

mysqlshow mysql

Mysqldump

Программа mysqldump используется для создания дампа содержания базы данных MySQL. Она пишет инструкции SQL в стандартный вывод. Эти инструкции SQL могут быть переназначены в файл. Можно резервировать базу данных MySQL, используя mysqldump, но при этом Вы должны убедиться, что в этот момент с базой данных не выполняется никаких других действий. А то mysqldump Вам такого нарезервирует...

Программа mysqldump поддерживает следующие параметры (Вы можете использовать короткую или подробную версию):

Таблица 10 - Опции команды MySQLdump

-#, --debug=[options]	**Вывести в протокол отладочную информацию. В общем виде 'd:t:o, filename`.**
-?, --help	Справка.
-c, --compleat-insert	Генерируйте полные инструкции insert (не исключая значений, которые соответствуют значениям столбца по умолчанию).
-h, --host=[hostname]	Соединиться с сервером hostname.
-d, --no-data	Экспорт только схемы информации (исключая данные).
-t, --no-create-info	Экспорт только данных, исключая информацию для создания таблицы. Противоположность -d.
-p, --password=[password]	Пароль пользователя, для соединения с

	сервером MySQL. Обратите внимание, что не должно быть пробела между -p и паролем.
-q, --quick	Не буферизовать результаты запроса, дамп выдать непосредственно к STDOUT.
-u, --user=[username]	Имя пользователя. Если не задано, используется текущий логин.
-v, --verbose	Вывести подробную информацию относительно различных стадий выполнения mysqldump.
-P, --port=[port]	Порт для связи.
-V, --version	Информация о версии.

Вы можете направить вывод mysqldump в клиентскую программу MySQL, чтобы копировать базу данных. ПРИМЕЧАНИЕ: Вы должны убедиться, что база данных не изменяется в это время, иначе Вы получите противоречивую копию!

Для справки:

mysqldump -u root -p mysql user>mysql-1.sql

mysqldump -u root mysql>mysql-2.sql

Примечание флаг –p используется в случае, если пользователь наделен паролем.

После выполнения этой команды у нас появился файл mysql-1.sql и mysql-2.sql. Загрузим их в текстовый редактор, чтобы поподробнее изучить, и, возможно, немного поправить.

Задание

Запустите сервер MySQL. Зарегистрируйте своего пользователя в консольном приложении, задайте ему права.

С помощью утилиты Mysqlshow выполните команду на просмотр структуры и состав таблиц базы Mysql. Приведите в отчете её схему. С помощью

утилиты Mysqldump получите полный дамп базы Mysql (данные и таблицы), а также отдельные дампы таблиц и данных.

Контрольные вопросы:

1. Каким способом возможен запуск серверной части СУБД.
2. Что такое привилегия. Каково её предназначение.
3. Какие основные утилиты входят в состав СУБД, какие функции они выполняют.

Практическая работа №9

по учебной дисциплине **«Защита информации в системах управления»**

Раздел 2. Средства реализации защиты в информационных системах

Политика IP-безопасности

1. Цель работы

Изучить способы первичной защиты компьютера, протокол IPsec, принципы защиты электронной почты. Освоить технологию настройки политики IP-безопасности, настройки фильтрация TCP/IP

2. Подготовка к занятию

1. Изучить (повторить) теоретический материал.
2. Ознакомиться с программой лабораторной работы.
3. Подготовить отчет о лабораторной работе.
4. Ответить на контрольные вопросы.

3. Распределение времени занятия:

Всего: 90 мин
Вступительная часть 2 мин
Проверка готовности студентов к занятию 5 мин
Основная часть – 70 мин
Проверка выполнения практического занятия 10 мин
Заключительная часть 3 мин

4. Правила работы в лаборатории

К работе в лаборатории допускаются лица, изучившие правила и меры безопасности, сдавшие зачет по ним и усвоившие порядок выполнения лабораторной работы.

4.1. Требования безопасности перед началом работ

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории.

- Убедиться в целостности электрических розеток и разъемов. В лаборатории необходимо быть в сменной обуви.

- Включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

4.2. Требования безопасности во время работы

- выполняя практическое занятие, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и носители информации, непосредственно относящиеся к данному лабораторному занятию;
- подключение и отключение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса;
- при обнаружении неисправностей в оборудовании немедленно отключить источники питания и доложить об этом руководителю занятий или лаборанту.

4.3. Требования безопасности по окончании работы

- доложить руководителю занятий или лаборанту о завершении работ;
- привести в порядок и сдать рабочее место лаборанту, и доложить руководству

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Сети TCP/IP при отсутствии системы защиты могут быть подвергнуты многочисленным атакам, выполняемым как изнутри локальной сети, так и извне, если локальная сеть имеет соединение с глобальной сетью, например с Интернетом. Некоторые атаки носят пассивный характер и сводятся к мониторингу информации, циркулирующей в сети, другие активный, направленный на повреждение или нарушение целостности информации или самой сети. Наиболее распространенные типы вторжения на сети TCP/IP следующие:

- Подслушивание. Эти атаки используют уязвимость сети к перехвату сетевых пакетов специальными аппаратными и программными средствами. Если передаваемая информация не зашифрована, ее конфиденциальность будет нарушена.

- Искажение данных. В зависимости от своих целей злоумышленник, перехвативший сетевые данные, может модифицировать их и отправить по назначению, причем сделать это скрытно от отправителя и получателя.
- Фальсификация IP-адреса. В сети TCP/IP хост идентифицируется своим IP-адресом, указанным в IP-пакете, который несложно подделать. Такая подмена IP-адресов может выполняться с различными целями, например с целью сокрытия источника сообщения или для некорректной идентификации отправителя, позволяющей получить доступ к сетевым ресурсам.
- Подбор паролей. Получив пароль учетной записи, злоумышленник получает все права доступа легитимного пользователя и, если права достаточны, может сделать с системой что угодно.
- Атака DoS (Denial of Service - отказ в обслуживании). Заключается в создании препятствий в работе системы, что приводит к отказу от обслуживания обычных пользователей сети. Примером является направление на атакуемый сервер большого числа пакетов, перегружающих сетевой трафик.
- Компрометация ключей. Шифрование передаваемых по сети данных компьютера выполняют с помощью ключей, зависящих от применяемых криптографических средств. Поэтому раскрытие ключа шифрования означает потерю конфиденциальности передаваемой по сети информации. При этом злоумышленник сможет знакомиться с передаваемыми сообщениями и/или модифицировать их для достижения своих целей.
- Атака на прикладном уровне. Такие атаки выполняются с целью получения контроля над приложением, запущенным на сетевом компьютере. Например, хакер может попытаться получить доступ к приложению удаленного администрирования компьютером и в случае успеха получить возможность сделать с ним все, что угодно.

Для защиты ото всех этих атак были разработаны средства IP-безопасности, обеспечиваемые протоколом IPsec (Internet Protocol Security -протокол безопасности Интернета), представляющие собой набор открытых стандартов защиты соединений по IP-сетям средствами криптографии. Протокол IPsec нацелен на защиту пакетов, передаваемых по сетям TCP/IP.

Обзор IPsec

Протокол IPsec опирается на концепцию защиты, исходящей из предположения, что среда передачи данных не защищена. Сетевые компьютеры, пересылающие пакеты IPsec от источника к получателю, не имеют никаких сведений об использовании протокола IPsec и могут в принципе его не поддерживать. Таким образом, протокол IPsec может быть использован в локальных сетях с одноранговой и клиент-серверной организацией для передачи данных между маршрутизаторами и шлюзами глобальных сетей или в удаленных соединениях и частных сетях Интернета.

Протокол IPsec позволяет преодолеть ограниченность обычных средств защиты, полагающихся на защиту периметра локальной сети, брандмауэры, защищенные маршрутизаторы, средства аутентификации пользователей удаленного доступа. Защиту от внутренних атак указанные средства не обеспечивают, поскольку основаны на именах и паролях Учетных записей пользователей. Ясно, что защита периметра сети никак не воспрепятствует злоумышленнику, имеющему локальный доступ к компьютеру, с помощью различных программ извлечь из него все пароли учетных записей и далее их использовать для своих целей.

С другой стороны, ограничение физического доступа к оборудованию локальной сети часто невозможно, поскольку кабели локальной сети могут иметь большую протяженность и располагаться в местах, препятствующих

эффективной защите. Протокол IPsec позволяет преодолеть все эти проблемы, при его использовании компьютер шифрует все отправленные данные, а получатель - дешифрует. Поэтому при условии построения многоуровневой системы защиты, включающей ограничение физического доступа к компьютерам (но не линиям передачи данных), защиту периметра и корректную настройку пользовательского доступа, протокол IPsec беспечит всестороннюю защиту сетевых данных.

Протокол IPsec защищает не сам канал передачи информации, а передаваемые по нему пакеты; тем самым IPsec решает следующие задачи:

- Неотрицаемость сообщений. Протокол IPsec поддерживает создание цифровой подписи передаваемого сообщения закрытым ключом отправителя, что обеспечивает невозможность отрицания авторства сообщения.
- Аутентификация источника сообщения. Обеспечивается поддержкой инфраструктуры открытого ключа (PKI - Public Key Infrastructure), аутентифицирующей компьютер-отправитель на основе сертификата.
- Конфиденциальность передаваемых данных. Обеспечивается шифрованием информации криптостойкими алгоритмами DES и 3DES.
- Защита целостности данных. Осуществляется путем подписания передаваемых пакетов хеш-кодами аутентификации сообщения HMAC (Hash Message Authentication Codes). Коды HMAC вначале подсчитываются компьютером-отправителем сообщения, использующим специальный алгоритм и общий секретный ключ. Затем компьютер-получатель повторно подсчитывает код HMAC и сравнивает результат с полученным значением. Для подсчета HMAC используются криптостойкие алгоритмы MD5 и SHA.
- Защита от повторного использования перехваченных пакетов с целью получения доступа к ресурсам. Для управления средствами защиты IPsec применяются правила политики IP-безопасности, что значительно

упрощает развертывание IPsec на защищаемой системе. Политика IPsec применяется к локальным компьютерам, к домену и организационным подразделениям, созданным в активном каталоге. При настройке политики IPsec следует учесть правила безопасной работы, принятой в организации. Для этого в каждой политике IP-безопасности содержится несколько правил, применяемых к группам компьютеров или организационным подразделениям.

Установка политики IP-безопасности

Политику IP-безопасности можно установить для локального компьютера, домена или для всех доменов активного каталога. В системе Windows 7 политика IP-безопасности устанавливается с помощью оснастки Управление политикой безопасности IP в консоли MMC. Для открытия этой оснастки выполните следующие шаги:

1. Щелкните указателем мыши на кнопке Пуск(Start), на экране появится главное меню Windows 7. В строке ввода команд введите ттс и нажмите Ввод, отобразится диалог консоли MMC.
2. Щелкните на меню Файл(File) и выберите пункт Добавить или удалить оснастку.(Add/Remve Snap-in).
3. Из списка оснасток выбрать Управление политикой безопасности(IP Security Policy Management) и нажать кнопку Добавить(Add).

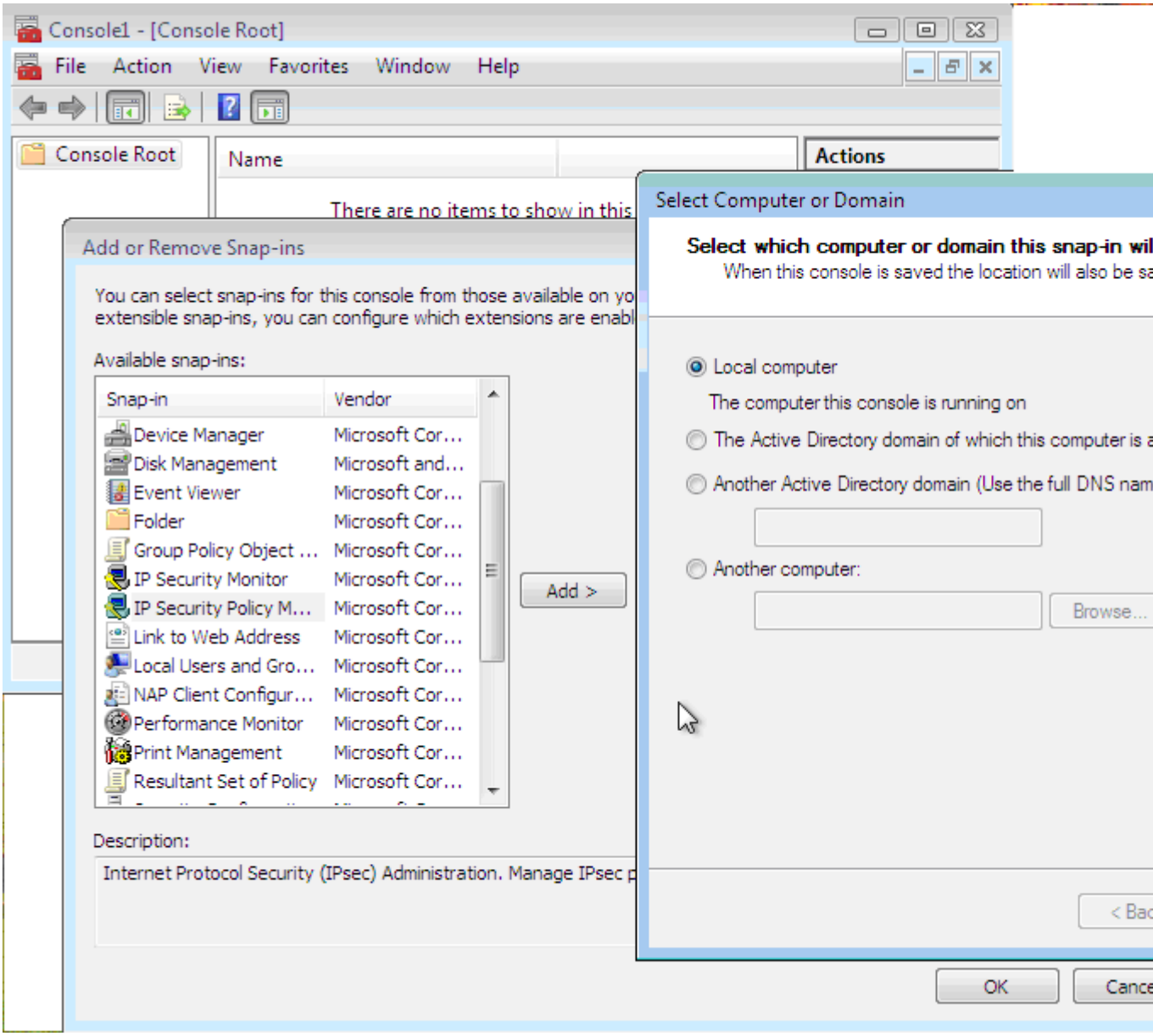


Рисунок 1 - Выбор оснастки управления политикой безопасности IP (экранная копия)

4. В окне Выбор компьютера или домена(Select Computer or Domain) выберите Локальный компьютер(Local computer) и нажмите кнопку Готово(Finish).

5. Затем нажмите ОК в окне Добавить или Удалить Оснастоки(Add or Remove Snap-in). По умолчанию в Windows 7 нет установленных политик IP- безопасности. Windows 7 позволяет создавать новую

политику IP-безопасности, для чего необходимо воспользоваться мастером установки политики IP-безопасности .

6. Щелкните правой кнопкой мыши на записи Политика безопасности IP (IP Security Policies on Local Computer) в левой части консоли MMC и выберите в контекстном меню команду Создать политику безопасности IP (Create Ip Security Policy). Эта команда отображает первый информационный диалог мастера политики IP-безопасности, щелкните на кнопке Далее (Next). В отобразившемся диалоге мастера укажите название создаваемой политики и укажите ее назначение, щелкните на кнопке Далее.

7. В отобразившемся диалоге укажите, должна ли новая политика использоваться по умолчанию. Оставьте отметку флажка Использовать правило по умолчанию и щелкните на кнопке Далее. В следующем диалоге мастера следует выбрать метод проверки подлинности подсоединяемого компьютера. Мастер позволяет выбрать только один из методов аутентификации:

- Стандарт службы каталогов (Протокол Kerberos V5). Протокол Kerberos представляет собою стандартную технологию защиты системы Windows 7. Этот метод годится для аутентификации всех компьютеров, поддерживающих протокол Kerberos V5 и входящих в доверяемый домен.
- Использовать сертификат данного центра сертификации (ЦС). В этом методе для аутентификации используются сертификаты. Применение данного метода требует предъявления сертификата открытого ключа, подтверждающего его подлинность. Для применения такого метода необходимо настроить хотя бы один центр сертификации CA (Certificate Authority). Этот метод идентификации используют для

доступа к Интернету или для доступа к компьютерам, на которых не установлен протокол Kerberos V5.

- Использовать данную строку для защиты обмена ключами. В этом методе аутентификация проводится на основе предварительно оговоренного секретного ключа. В качестве ключа можно употреблять слово, фразу, заранее оговоренные обоими пользователями. Использование этого метода не рекомендуется, так как ключ хранится в политике IPsec в открытом виде и для его защиты следует ограничить доступ к политике IPsec.

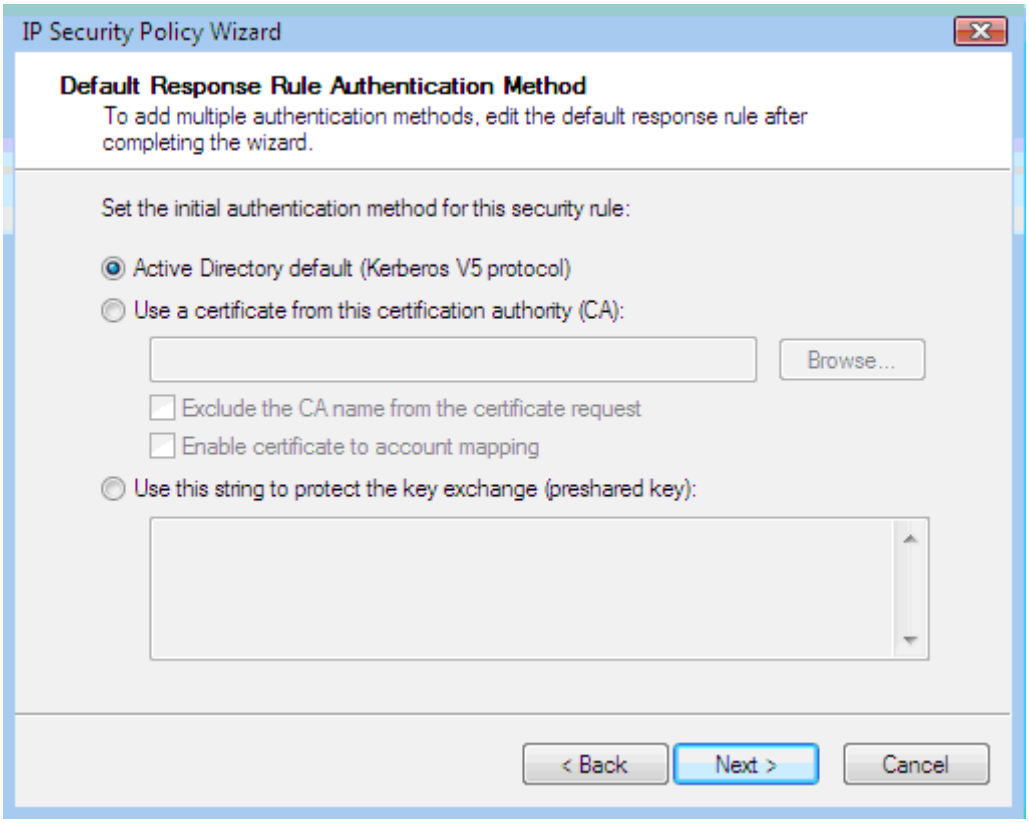


Рисунок 2 - Методы аутентификации (экранная копия)

8. Выберите метод и щелкните на кнопке Далее, отобразится завершающий диалог мастера с флажком, предлагающим изменить политику безопасности. Оставьте флажок на месте и щелкните на кнопке Готово. Отобразится диалог настройки свойств новой политики IP-безопасности .

Для изменения настроек политики IP-безопасности можно либо щелкнуть на кнопке Изменить и откорректировать параметры в отобразившемся диалоге, либо воспользоваться мастером, который автоматически запускается при отмеченном флажке Изменить политику безопасности в завершающем диалоге мастера политики IP-безопасности; это позволяет упростить процедуры настройки, которые достаточно сложны и требуют понимания принципов функционирования политики IP-безопасности.

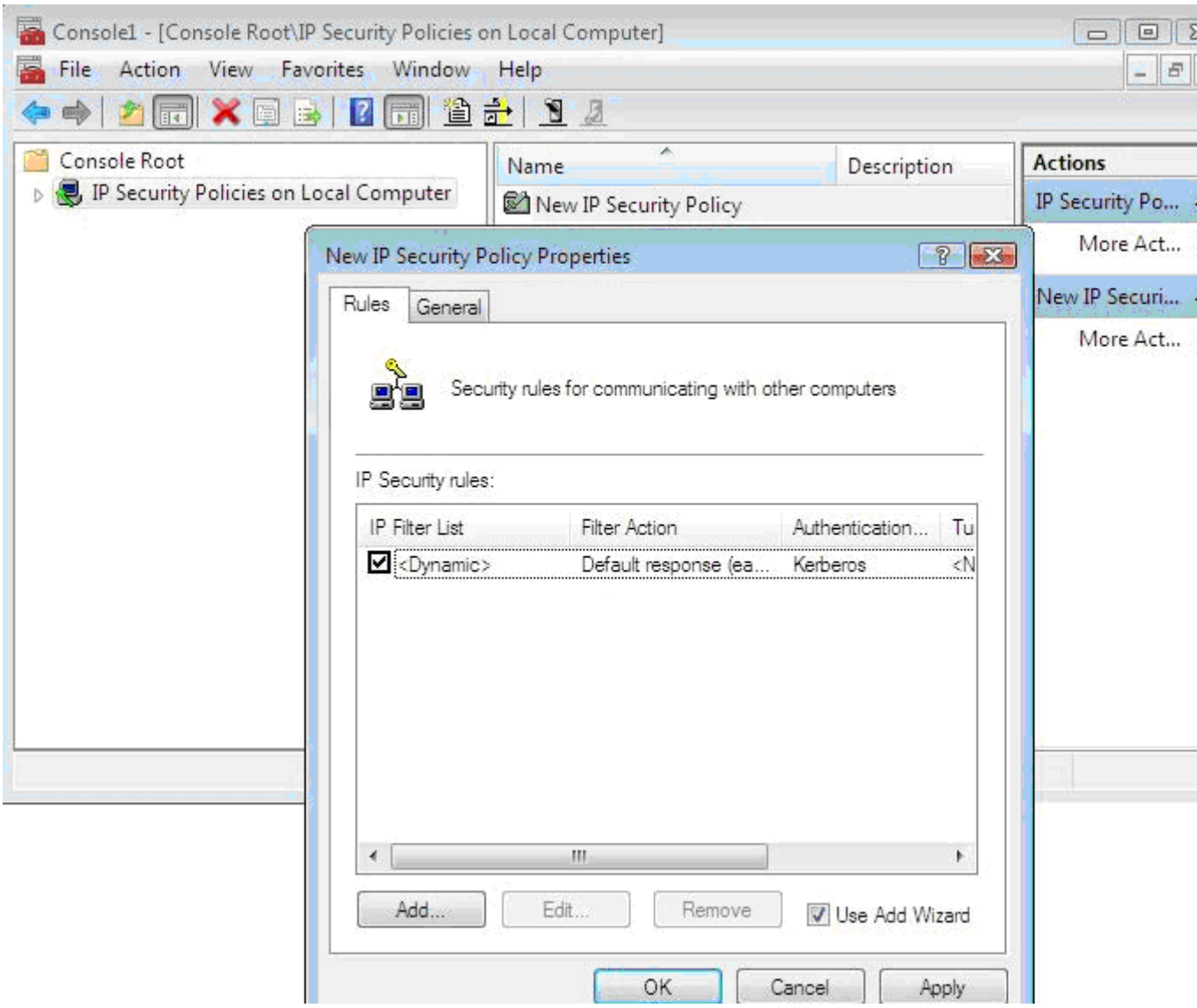


Рисунок 3 - Диалог настройки параметров политики IP-безопасности (экранная копия)

Назначение параметров настройки брандмауэра Windows

Работа брандмауэра Windows определяется параметрами: Включить; Включить, но не разрешать исключения и Выключить.

- Включить. По умолчанию брандмауэр включен, и если нет другого брандмауэра, то лучше оставить его в таком состоянии. В этом состоянии брандмауэр Windows будет блокировать все непредусмотренные запросы на подключение к вашему компьютеру за исключением тех, которые предназначены для программ или служб, выбранных на вкладке Исключения.

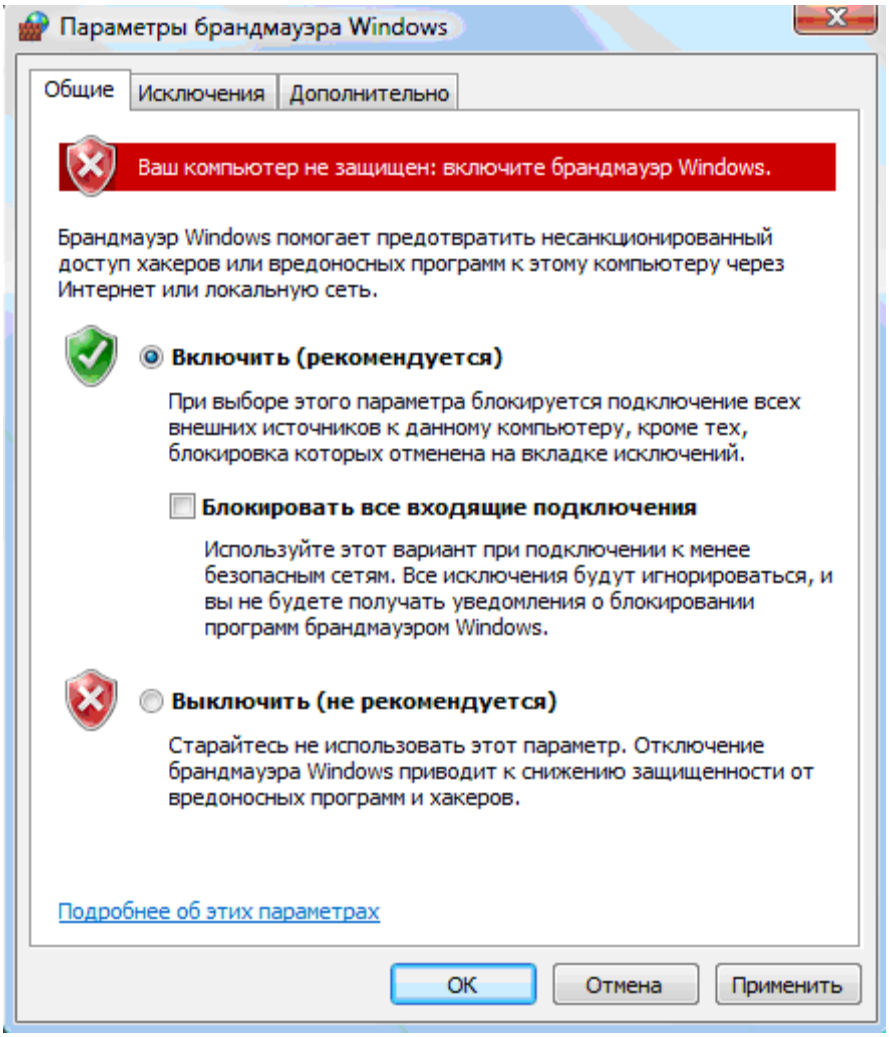


Рисунок 4 - Графический интерфейс брандмауэра Windows (экранная копия)

- Включить, но не разрешать исключения. При установке флажка Не разрешать исключения брандмауэр Windows блокирует все непредусмотренные запросы на подключение к компьютеру, в том числе и те, которые предназначены для программ и служб, выбранных на вкладке Исключения. Этот параметр служит для максимальной защиты компьютера, например при подключении к общедоступной сети в отеле или в аэропорту или в периоды распространения через Интернет особо опасных вирусов или червей. Нет необходимости все время использовать флажок Не разрешать исключения, поскольку при этом некоторые программы могут перестать работать правильно, а кроме того, будут блокироваться непредусмотренные запросы к следующим службам:

  - Служба доступа к файлам и принтерам;
  - Средства «Удаленный помощник» и «Дистанционное управление рабочим столом»;
  - Обнаружение сетевых устройств;
  - Заранее настроенные программы и службы в списке Исключения;
  - Дополнительные объекты, добавление в список Исключения.

Если установлен флажок Не разрешать исключения, можно по-прежнему отправлять и получать электронную почту, использовать программу передачи мгновенных сообщений или просматривать большинство Web-страниц.

- Выключить. Данный параметр отключает брандмауэр Windows, в результате чего компьютер становится уязвим к атакам злоумышленников или вирусов. Этот параметр могут использовать только опытные пользователи временно в целях администрирования компьютера или при наличии защиты другим брандмауэром. Параметры настройки, заданные для случая, когда компьютер

подсоединен к домену, сохраняются отдельно от параметров для работы компьютера не в составе домена. Эти отдельные группы параметров настройки называются профилями.

Чтобы включить или выключить брандмауэр Windows, необходимо

1. Войти в систему под учетной записью Администратор;
2. В меню Пуск выбрать команды Настройка и Панель управления;
3. Дважды щелкнуть на значке Брандмауэр Windows;
4. На вкладке Общие выбрать один из следующих параметров:

- Включить(рекомендуется) ;
- Выключить (не рекомендуется).

Помогая обеспечить защиту компьютера, брандмауэр Windows блокирует непредусмотренные запросы на подключение к вашему компьютеру. Поскольку брандмауэр ограничивает обмен данными между компьютером и Интернетом, может потребоваться регулировка параметров для некоторых программ, которым требуется свободное подключение к Интернету. Для этих программ можно сделать исключение, чтобы они могли связываться через брандмауэр.

Однако следует помнить, что каждое исключение, дающее программе возможность связываться через брандмауэр Windows, делает компьютер уязвимым. Создание исключения равносильно пробиванию бреши в брандмауэре.

Если таких брешей окажется слишком много, брандмауэр уже не будет прочной преградой. Обычно взломщики используют специальные программы для поиска в Интернете компьютеров с незащищенными подключениями. Если создать много исключений и открыть много портов, компьютер может оказаться жертвой таких взломщиков.

Чтобы уменьшить потенциальный риск при создании исключений:

- создавайте исключение, только когда оно действительно необходимо;
- никогда не создавайте исключений для незнакомой программы;
- удаляйте исключения, когда необходимость в них отпадает.

Иногда требуется открыть кому-то возможность связи с вашим компьютером, несмотря на риск, - например, когда ожидается получение файла, посланного через программу передачи мгновенных сообщений, или когда хочется принять участие в сетевой игре через Интернет.

Если идет обмен мгновенными сообщениями с собеседником, который собирается прислать файл (например, фотографию), брандмауэр Windows запросит подтверждение о снятии блокировки подключения и разрешении передачи фотографии на ваш компьютер.

Чтобы разрешить непредусмотренные подключения к программе на своем компьютере, в брандмауэре Windows используется вкладка Исключения.

Если программа или служба, для которых требуется создать исключение, отсутствуют в списке на вкладке Исключения, можно добавить ее с помощью кнопки Добавить программу. Если программа отсутствует в списке программ, которые можно добавить, нажмите кнопку Обзор, чтобы найти ее, затем откройте интерфейс брандмауэра Windows и на вкладке Исключения в группе Программы и службы установите флажок для программы или службы, которые требуется разрешить, и нажмите кнопку ОК.

Если программа или служба, которые требуется разрешить, отсутствуют в списке, выполните следующие действия: нажмите кнопку Добавить программу. В диалоговом окне Добавление программы выберите необходимую программу и нажмите кнопку ОК. Эта программа появится (с установленным флажком) на вкладке Исключения в группе Программы и службы. Нажмите кнопку ОК.

Если программа или служба, которые требуется разрешить, не указаны в перечне диалогового окна Добавление программы, выполните следующие действия: в диалоговом окне Добавление программы нажмите кнопку Обзор, найдите программу, которую требуется добавить, и дважды щелкните ее. Программа появится в группе Программы в диалоговом окне Добавление программы. Нажмите кнопку ОК. Эта программа появится (с установленным флажком) на вкладке Исключения в группе Программы и службы. Нажмите кнопку ОК.

Для некоторых программ номера портов не определены заранее. Эти программы открывают порты автоматически при необходимости. Чтобы такие программы могли соединиться с вашим компьютером, брандмауэр Windows должен позволить программе открыть нужный порт. Для правильной работы таких программ их необходимо внести в список на вкладке Исключения брандмауэра.

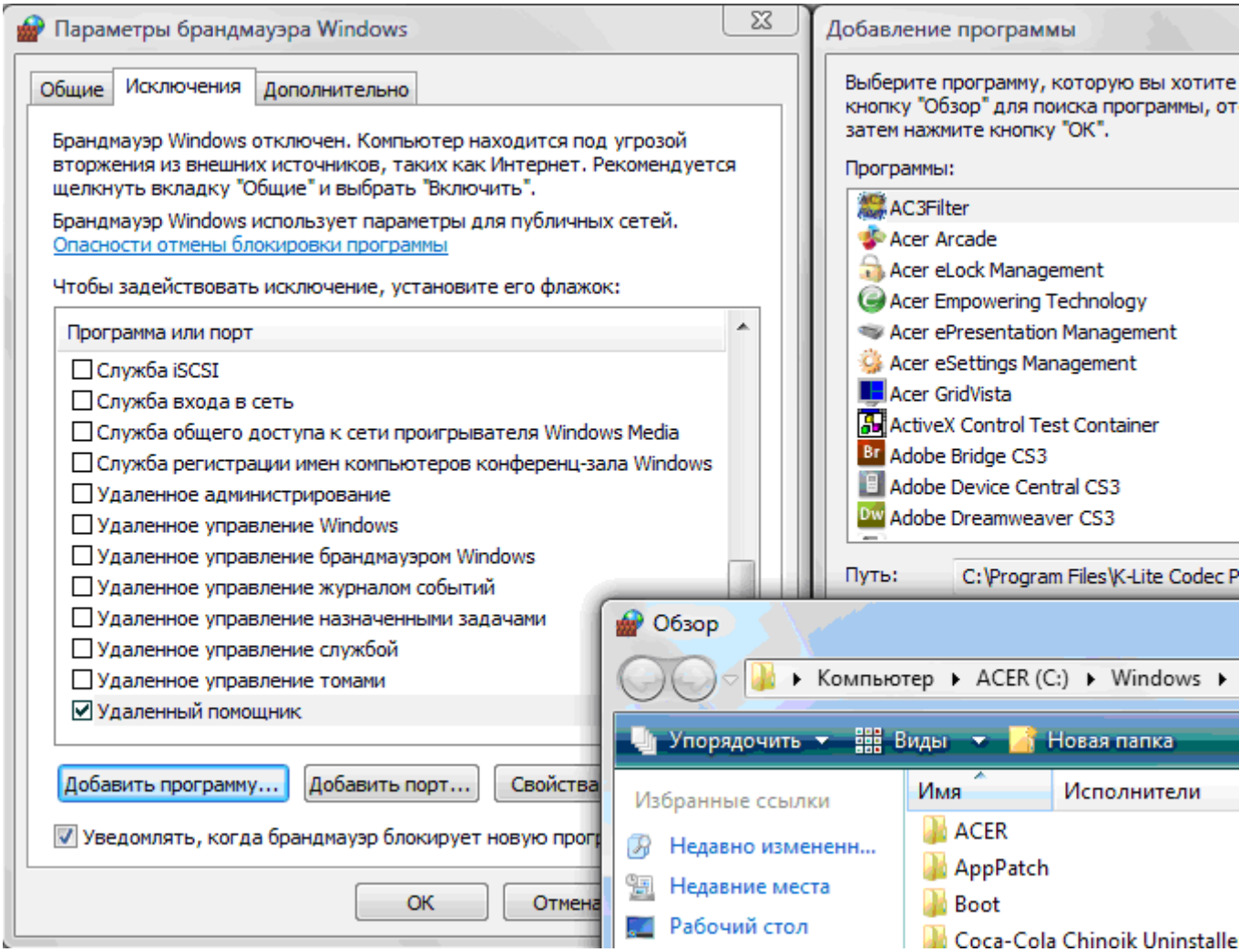


Рисунок 5 - Подключение непредусмотренных программ и служб (экранная копия)

Чтобы обеспечить безопасность компьютера, необходимо держать брандмауэр Windows (или другой брандмауэр по выбору) включенным, чтобы он блокировал любые непредусмотренные запросы на подключение к компьютеру. Для возможности подключения такого типа необходимо разрешить исключение или открыть порт для конкретной программы или службы. Порт - это проход в ваш компьютер, через который может передаваться информация. Если идет обмен мгновенными сообщениями с собеседником, который собирается прислать файл, брандмауэр Windows запросит подтверждение о снятии блокировки подключения и разрешение передачи этого файла на ваш компьютер. При желании участвовать в сетевой

игре через Интернет с друзьями вы можете добавить эту игру как исключение, чтобы брандмауэр пропускал игровую информацию на ваш компьютер.

Каждый открытый порт, дающий программе возможность связываться через брандмауэр Windows, делают компьютер уязвимым. Открытие порта также равносильно пробиванию бреши в брандмауэре. Если открыть много портов, компьютер может оказаться жертвой взломщиков. Чтобы уменьшить потенциальный риск при открытии портов:

- открывайте порт, только когда он действительно необходим;
- никогда не открывайте порт для программы, которую плохо знаете;
- закрывайте порт, когда необходимость в нем отпадает.

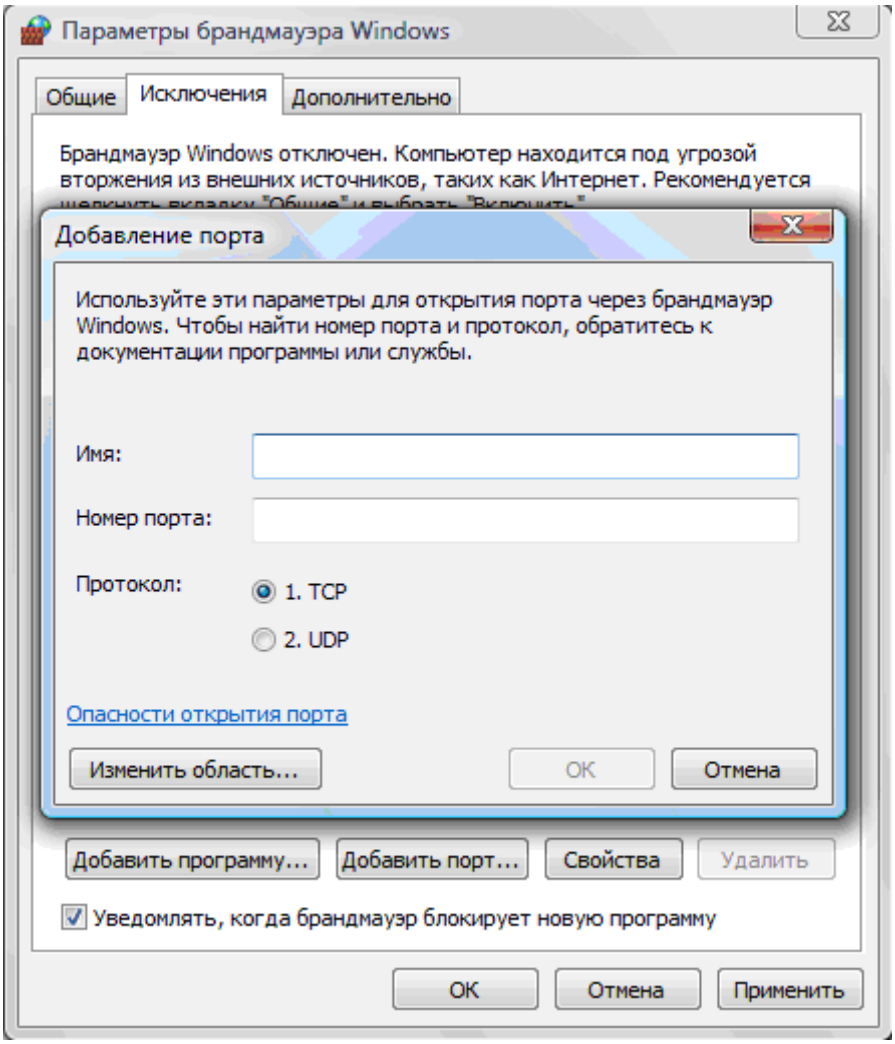


Рисунок 6 - Разрешение работы программ через порт (экранная копия)

У каждого порта есть номер, который играет роль адреса. Многие программы и службы имеют «постоянные адреса», т. е. для них заранее определены номера портов. Номера портов, необходимые для программы или службы, можно найти в документации производителя или на Web-узле.

Если программу не удалось найти, можно открыть порт. Чтобы определить, какой порт нужно открыть, на вкладке Исключения нажмите кнопку Добавить порт.

При добавлении или изменении параметров настройки для службы или

программы, например для игры, следует выбрать условие открытия порта: для любого компьютера или только для компьютеров локальной сети.

Если выбран вариант Любой компьютер, тогда к компьютеру сможет подключиться любой компьютер из Интернета или из локальной сети. Если выбран вариант Только локальная сеть, к данному компьютеру смогут подключиться компьютеры только из локальной сети.

Защита электронной почты

Для защиты своей электронной почты учтите, что именно вы и никто другой представляет наибольшую опасность для своей переписки. Именно вы пересылаете по почте все, что угодно, в открытом виде; именно вы щелкаете на всяких вложениях и отвечаете на самые конфиденциальные вопросы, в том числе о своей учетной записи, в ответ на послания в ваших почтовых ящиках; именно вы раздаете направо и налево свои идентификационные данные; наконец, именно вы никак не хотите прислушаться к совету -

перевести свою переписку на Web-сайты, предоставляющие почтовые услуги.

Запомните, что, если вы используете почтового клиента, никогда не конфигурируйте его на автоматическое открытие почтовых вложений. Любое послание от любого лица может содержать вложение самого опасного характера, поскольку его может послать кто угодно, в том числе вирус, заразивший компьютер отправителя. Следует инсталлировать программу антивируса и установить режим непрерывного сканирования поступающей почты, непрерывно обновляя свои антивирусные средства. Если же вам требуется просмотреть подозрительное вложение, то поместите присланный файл на отдельный диск (дискету) и проверьте его на вирус.

Для защиты электронных посланий от фальсификации в современном информационном сообществе применяется метод электронной подписи. В РФ принят закон об электронной цифровой подписи, как и во многих других странах. Легализация своей электронной подписи - дело не бесплатное, но если вы используете электронную почту для деловой переписки, то верхом неосторожности было бы рассылать письма без электронной подписи, да еще и в открытом виде.

Если у Вас нет настоящей, юридически легитимной цифровой подписи, используйте хотя бы программу PGP, и обменяйтесь со своими деловыми партнерами подписанными PGP-ключами.

Перехват паролей учетной записи почтового сервера - это серьезная проблема. Существуют почтовые протоколы, например APOP (это протокол POP, дополненный аутентификацией клиента) и SASL (Simple Authentication and Security Layer - уровень простой аутентификации и защиты), защищающие связь клиента с почтовым сервером от перехвата паролей. Но для их использования требуется поддержка протоколов APOP и SASL как

клиентом, так и сервером почтовой службы. Попробуйте выяснить, так это или нет, у своего Интернет-провайдера.

Всех этих проблем лишена почтовая служба, предоставляемая через Web. Вместо настройки почтового клиента, что вовсе не так просто, как кажется на первый взгляд, вы сгружаете на Web-сайте регистрационную Web-страницу, где указываете свое входное имя и пароль, а также некоторую другую информацию. Помните, что в ответ на просьбу указать свой домашний адрес, телефон, имя и фамилию и т. д. не следует указывать реальные данные. Далее вы щелкаете на кнопке - и для вас создается почтовый ящик на заокеанском сервере, принадлежащий солидной фирме, которая не будет продавать ваши электронные адреса всяким спаммерам.

Как же можно отличить солидную фирму ото всех прочих? Если вы работаете с Web-сайтом, предлагающим почтовый сервис, то настоящие сайты все пересылки конфиденциальной информации выполняют в защищенном режиме, использующем сокеты SSL (Secure Sockets Layer - протокол защищенных сокетов). При работе с сервером, поддерживающим сокеты SSL, в интернет-адресе сайта вместо записи http:// появляется запись https:// (HyperText Transmission Protocol Secure - протокол защищенной передачи гипертекстов), а в строке состояния браузера IE отображается замочек. Щелчок на замочке открывает диалог с сертификатом Web-сайта, где вы сразу можете увидеть, кто владелец этого Web-сайта - известная фирма Microsoft или подозрительная компания. Без такой проверки связываться с почтовым сервером вряд ли стоит - вы просто не будете знать, куда шлете письма и что там с ними будут делать. Более того, ваши пароли и имена, вводимые при регистрации, будут в открытом виде долго путешествовать по проводам, где любитель чужих секретов без проблем извлечет их из общего потока информации и использует по своему усмотрению.

1. Зарегистрируйтесь на надежном, сертифицированном Web-сайте, предоставляющем почтовые услуги, например http://www.gmail.com, которые к тому же выполняют антивирусную проверку поступающей почты.
2. Зашифруйте свое сообщение с помощью открытого PGP-ключа получателя, который вы лично от него получили и подписали своей цифровой подписью, т.е. подтвердили достоверность ключа. То же самое сделайте с почтовыми вложениями, причем это даже выгодно с точки зрения затрат времени, поскольку PGP-шифрование сжимает данные. Теперь любому злоумышленнику придется туго, поскольку взломать PGP-ключ длиной 2 Кбайт (2048 бит) может быть по силам только достаточно мощной организации. Но следует заметить, что, если на вашем компьютере «работает» клавиатурный шпион, все ваши хлопоты по поводу безопасности электронной почты пойдут насмарку.

В этом случае все ваши пароли, явки и адреса злоумышленник, установивший в ваш компьютер «жучка», будет знать не хуже вас. Так что не забудьте перед настройкой безопасности почтовой службы вначале проверить свой компьютер на предмет наличия «Троянов» и клавиатурных шпионов. И учтите, что эти шпионы без труда могут быть установлены, пока вы где-то ходите, не запустив парольную заставку, или не выключили компьютер.

Правила Интернет – безопасности

Чтобы защититься от угроз, связанных с сервисами электронной почты и доставки файлов, следуйте таким рекомендациям:

- Установите на своем компьютере антивирусную программу и настройте ее на непрерывную проверку электронной почты и загружаемых файлов на наличие вирусов.
- Чтобы исключить перехват паролей доступа к электронной почте, воспользуйтесь почтовыми службами, предоставляемыми на Web-сайтах, поддерживающих SSL-доступ при регистрации на сайте и имеющих сертификат от доверенных бюро CA.
- Отсылаемые электронные послания шифруйте и подписывайте своей электронной подписью. Если у вас нет легитимной электронной подписи, то воспользуйтесь средствами PGP для создания доверенных отношений с получателями своих писем.
- Для защиты службы доставки файлов воспользуйтесь шифрованием, применяя его в том числе при передаче файлов через FTP-сервер. При шифровании средствами PGP применяйте подписанные PGP-ключи, достоверность которых вами установлена.
- Избегайте загрузки файлов с ненадежных Web-сайтов, в число которых следует автоматически отнести все сайты, не поддерживающие SSL-доступ с надежным сертификатом.
- Чтобы исключить потерю конфиденциальности своей работы на компьютере, тщательно следите за отсутствием на нем троянских коней и клавиатурных шпионов. Лучшее средство защиты - не допускать несанкционированную установку злоумышленной программы.
- Следите за состоянием своего компьютера, просматривая скрытые процессы, проверяя наличие неведомо откуда появившихся папок и файлов, особенно скрытых, и списки установленных программ. Для этого можно воспользоваться программами защиты от клавиатурных шпионов, например, программы Cleaner.
- Особо подготовленные пользователи могут просматривать содержимое системного реестра с целью выявления подозрительных записей. Всем

остальным можно воспользоваться специальными программами контроля содержимого системного реестра.

- Для пресечения работы клавиатурного шпиона воспользуйтесь программным брандмауэром, который закроет доступ шпиону к Интернету и выдаст его присутствие при попытке передать собранную информацию через электронную почту.

Практическая работа №10
по учебной дисциплине **«Защита информации в системах управления»**

Раздел 2. Средства реализации защиты в информационных системах

Тема 10. Защита от разрушающих программных воздействий. Использование антивирусных средств

1. Цель работы:

Изучение принципов работы антивирусных средств и их использования для зашиты от вредоносных программных воздействий. Работа выполняется

2. Подготовка к занятию

1. Изучить (повторить) теоретический материал.
2. Ознакомиться с программой лабораторной работы.
3. Подготовить отчет о лабораторной работе.
4. Ответить на контрольные вопросы.

3. Распределение времени занятия:

Всего: 90 мин
Вступительная часть 2 мин
Проверка готовности студентов к занятию 5 мин
Основная часть – 70 мин
Проверка выполнения практического занятия 10 мин
Заключительная часть 3 мин

4. Правила работы в лаборатории

К работе в лаборатории допускаются лица, изучившие правила и меры безопасности, сдавшие зачет по ним и усвоившие порядок выполнения лабораторной работы.

4.1. Требования безопасности перед началом работ

- ЗАПРЕЩАЕТСЯ: переодеваться, пользоваться огнем, курить, принимать пищу в лаборатории.
- Убедиться в целостности электрических розеток и разъемов. В лаборатории необходимо быть в сменной обуви.

- Включение компьютера производить только после получения допуска по выполняемой работе и разрешения преподавателя или лаборанта.

4.2. Требования безопасности во время работы

- выполняя практическое занятие, студенты обязаны использовать только вычислительную технику, периферийное оборудование, соединительные кабели, измерительное оборудование и носители информации, непосредственно относящиеся к данному лабораторному занятию;
- подключение и отключение составляющих вычислительного комплекса производить только при полном снятии напряжения со всех составляющих вычислительного комплекса;
- при обнаружении неисправностей в оборудовании немедленно отключить источники питания и доложить об этом руководителю занятий или лаборанту.

4.3. Требования безопасности по окончании работы

- доложить руководителю занятий или лаборанту о завершении работ;
- привести в порядок и сдать рабочее место лаборанту, и доложить руководству

1. Порядок выполнения лабораторной работы

Вариант задания выдается преподавателем и определяется файлом с тестовым вирусом. Необходимо ознакомится с интерфейсом пользователя, с основными параметрами настройки программы и настроить ее так, чтобы она обнаруживала тестовый вирус.

1.1 Изучение интерфейса пользователя

Интерфейс **Антивируса Касперского 6.0** состоит из четырех окон:

- **Главного окна**, в котором можно управлять задачами и компонентами антивируса. В нем также расположены ссылки на остальные окна
- **Окна настроек**, предназначенного для настройки задач и компонентов
- **Окна статистики и отчетов**, в котором можно получить данные о результатах работы антивируса

- **Окна справочной системы**

Дополнительно, **Антивирус Касперского** встраивается в контекстное меню объектов, размещенных на жестком диске, добавляет свою группу в системное меню **Пуск** и во время работы в системной панели операционной системы появляется иконка антивируса.

Для ознакомления с интерфейсом пользователя нужно будет поочередно вызвать все четыре окна интерфейса **Антивируса Касперского 6.0**.

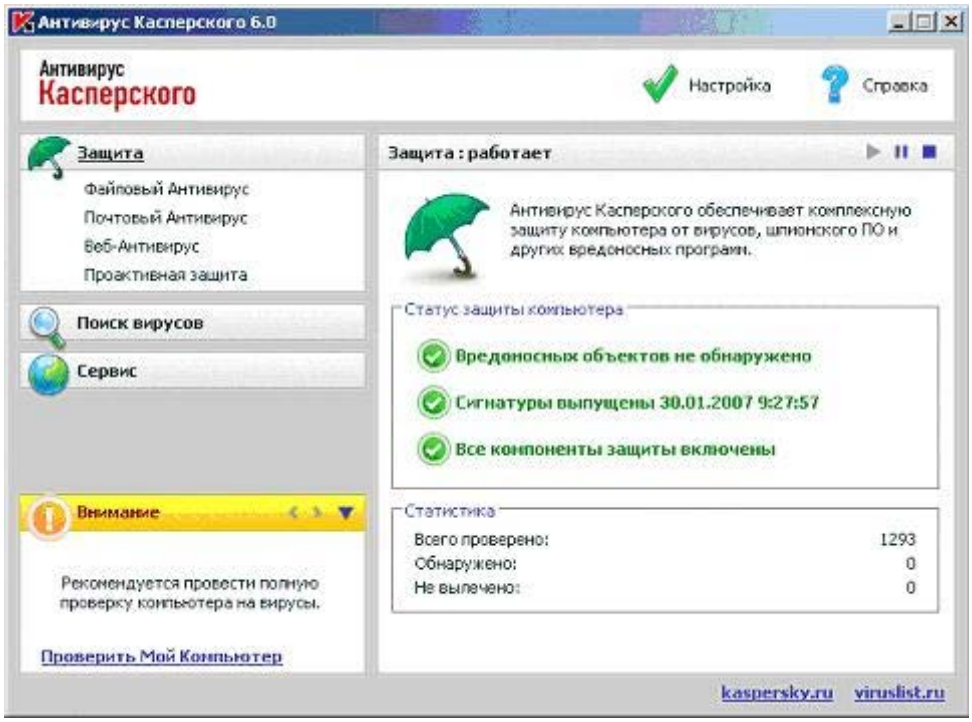


Рис. 1. Главное окно программы «Антивирус Касперского 6.0»

Антивирус Касперского 6.0 включает в себя три основные подсистемы:

– Антивирусный монитор: программа, следящая за обращением процессов к файлам и проверяющая процессы и файлы на предмет наличия вирусов. Антивирусный монитор работает постоянно в фоновом режиме.
– Антивирусный сканер: программа, которая может быть запущенная вручную пользователем или по определенному расписанию. Выполняет проверку различных ресурсов, таких как жесткие диски,

съемные диски, сетевые диски, отдельные папки в файловой системе, оперативную память, объекты автозапуска на предмет наличия вирусов.

- Сервисная подсистема: выполняет функции обновления базы вирусных сигнатур, создания аварийного диска, ведения отчетов о работе антивирусной защиты, резервного хранения объектов, хранения объектов, помещенных на карантин.

В терминах интерфейса пользователя эти функции располагаются в трех категориях:

1. Защита
2. Поиск вирусов
3. Сервис

В верхней правой части окна размещено две ссылки: **Настройка** и **Справка**. Первая используется для настройки антивируса, вторая - для вывода справочной системы.

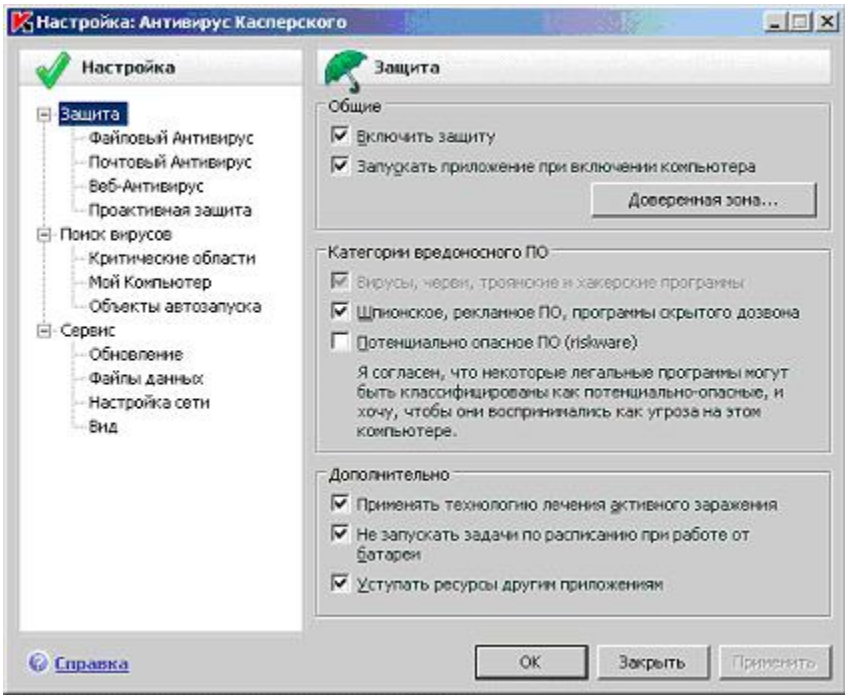


Рис. 2. Окно настроек программы

Антивирусный монитор обеспечивает защиту в режиме реального времени, т. е. постоянно проверяет файлы, которым происходит обращение. В терминах «Антивируса Касперского 6.0» такая функциональность носит название «защита» и делится на защиту файловой системы, проверку электронной почты (протоколы SMTP, POP3, IMAP), веб-антивирус (проверка HTTP трафика), проактивную защиту (противостояние неизвестным вирусам, контроль запуска программ, обращений к реестру Windows).

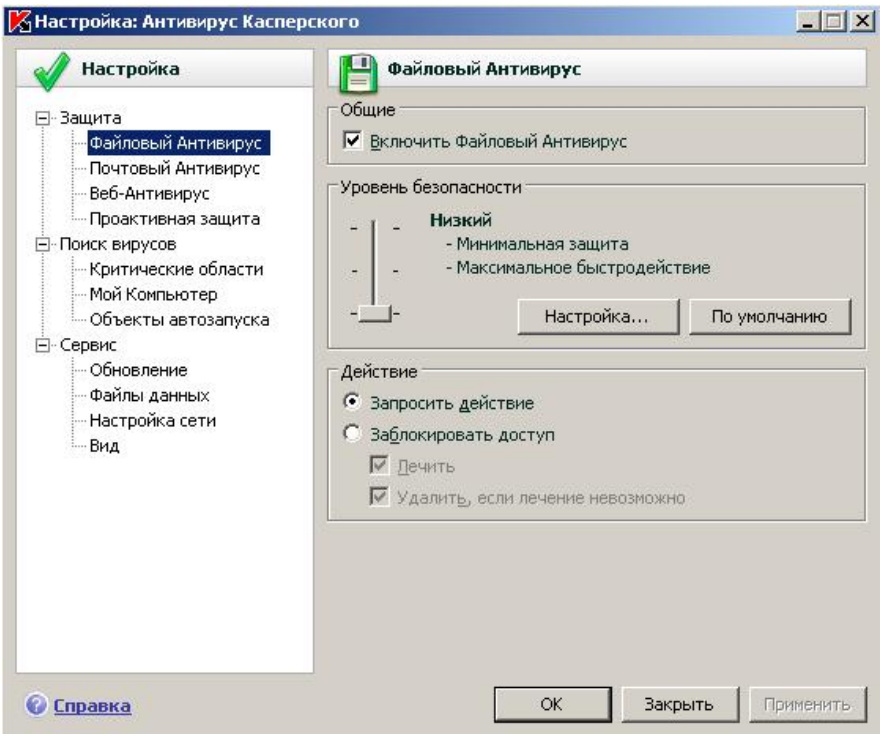


Рисунок 3. Окно настроек файлового антивируса

Антивирусный сканер (в терминах «Антивируса Касперского 6.0» - поиск вирусов) выполняет сканирование ресурсов компьютера в целях поиска вирусов. Сканирование может быть запущено пользователем вручную или по заранее установленному расписанию.

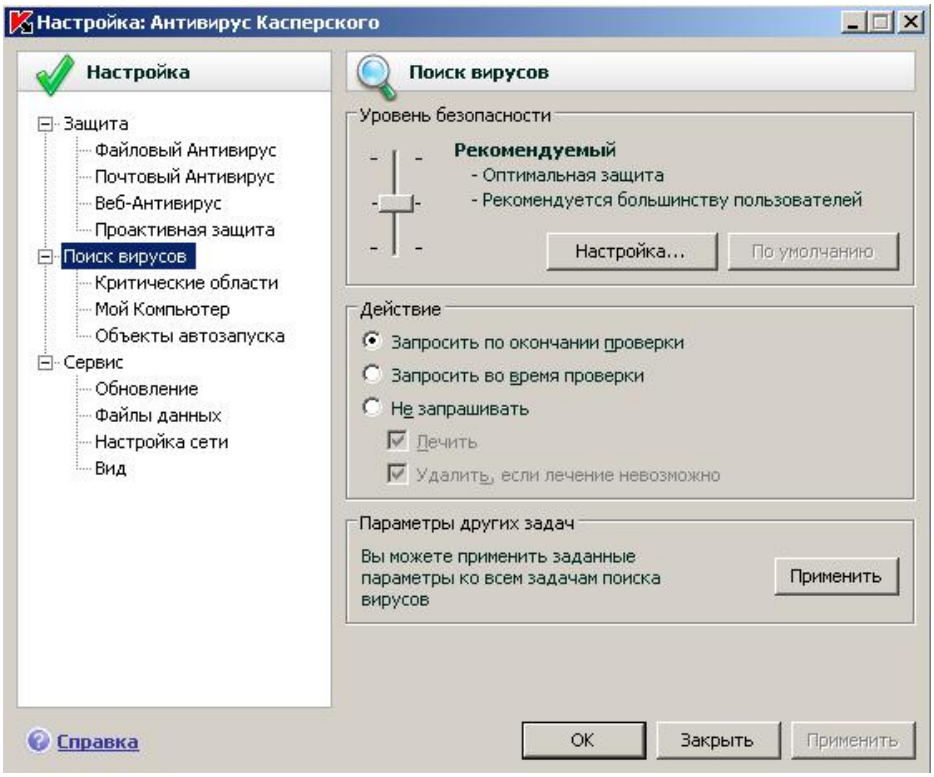


Рис. 4. Окно настроек поиска вирусов

В узле «Сервис» располагаются средства настройки обновления антивирусных баз, ведения файлов отчетов, параметров уведомлений, настройки сети и внешнего вида программы.

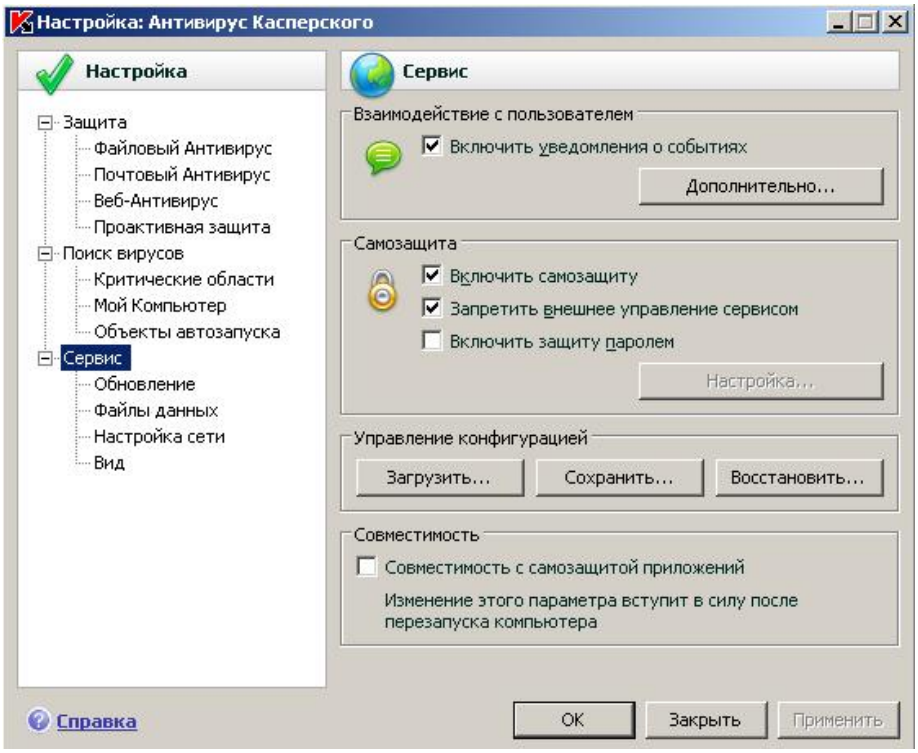


Рис. 5. Окно настроек сервисных функций

1.2 Тестирование антивирусной функциональности

1.2.1 Использование тестового вируса EICAR

Тестовый вирус EICAR (European Institute for Computer Antivirus Research) разработан Европейским институтом компьютерных антивирусных исследований.

EICAR – это небольшой 68 байтный файл, который при запуске на незащищенном компьютере вызывает показ уведомления "EICAR-STANDARD-ANTIVIRUS-TEST-FILE!". Иных, свойственных вирусам проявлений он не несет. Однако если на компьютере стоит и исправно работает антивирус, EICAR будет заблокирован. Это происходит потому, что все ведущие производители антивирусных программ договорились между собой - считать EICAR вирусом и применять к нему все правила и действия, применяемые к настоящим вредоносным программам.

Для создания антивируса необходимо открыть текстовый редактор и ввести следующую строку символов:

X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*

После этого следует сохранить файл с расширением .com.

Для более подробного тестирования можно применять другие расширения. Например, если указать .txt, можно проверить проверяются ли текстовые файлы. Для проверки будут ли обнаруживаться вирусы в архивах, EICAR можно заархивировать.

1.2.2 Модификация тестового вируса EICAR

Суть EICAR такова, что он оказывается неизлечимым. Это происходит потому, что антивирус идентифицирует EICAR как вирус по наличию в нем упомянутых 68 символов. Если их удалить - то от файла ничего не останется. Следовательно, с помощью EICAR можно тестировать только основную функцию антивируса - обнаружение.

1.2.3 Создать файл CURE-EICAR. Файл создается в текстовом редакторе путем добавления в начало вируса символов “CURE-” и сохранения файла с расширением .com. Обнаружив такой файл антивирус «вылечит» его, сократив размер файла до 4 байт (символы «CURE»).

Рис. 6. Модификация вируса CURE-EICAR

1.2.4 Создать файл DELE-EICAR. Файл создается в текстовом редакторе путем добавления в начало вируса символов “DELE-” и сохранения файла с расширением .com. Обнаружив такой файл, антивирус определяет его как неизлечимый или троянскую программу и удаляет. По результатам проверки файл должен остаться только в резервном хранилище.

Рис. 7. Модификация вируса DELE-EICAR

1.2.5 Создать файл CORR-EICAR. Файл создается в текстовом редакторе путем добавления в начало вируса символов “CORR-” и сохранения файла с расширением .com. Обнаружив такой файл, антивирус определяет его как файл с поврежденной структурой, вследствие чего проверить его на наличие вирусов невозможно. Такой файл признается условно чистым.

Рис. 8. Модификация вируса CORR-EICAR

1.2.6 Создать файл ERRO-EICAR. Файл создается в текстовом редакторе путем добавления в начало вируса символов “ERRO-” и сохранения файла с расширением .com. При сканировании такого файла, антивирус обнаружит ошибку при анализе его содержимого (например, при нарушении целостности при проверке многотомного архива). Такой файл признается условно чистым.

Рис. 9. Модификация вируса ERRO-EICAR

1.2.7 Создать файл SUSP-EICAR. Файл создается в текстовом редакторе путем добавления в начало вируса символов “SUSP-” и сохранения файла с расширением .com. При сканировании такого файла антивирус считает его подозрительным, а именно зараженным неизвестным вирусом. Такой файл должен быть помещен на карантин или удален.

Рис. 10. Модификация вируса SUSP-EICAR

1.2.8 Создать файл WARN-EICAR. Файл создается в текстовом редакторе путем добавления в начало вируса символов “WARN-” и сохранения файла с расширением .com. Такой файл также признается подозрительным, но не неизвестным вирусом, а модификацией известного.

1.3 Сохранить отчет с результатами работы антивируса в текстовый файл. Для этого в главном окне программы выбрать раздел «Защита» и в нем контейнер «Статистика»

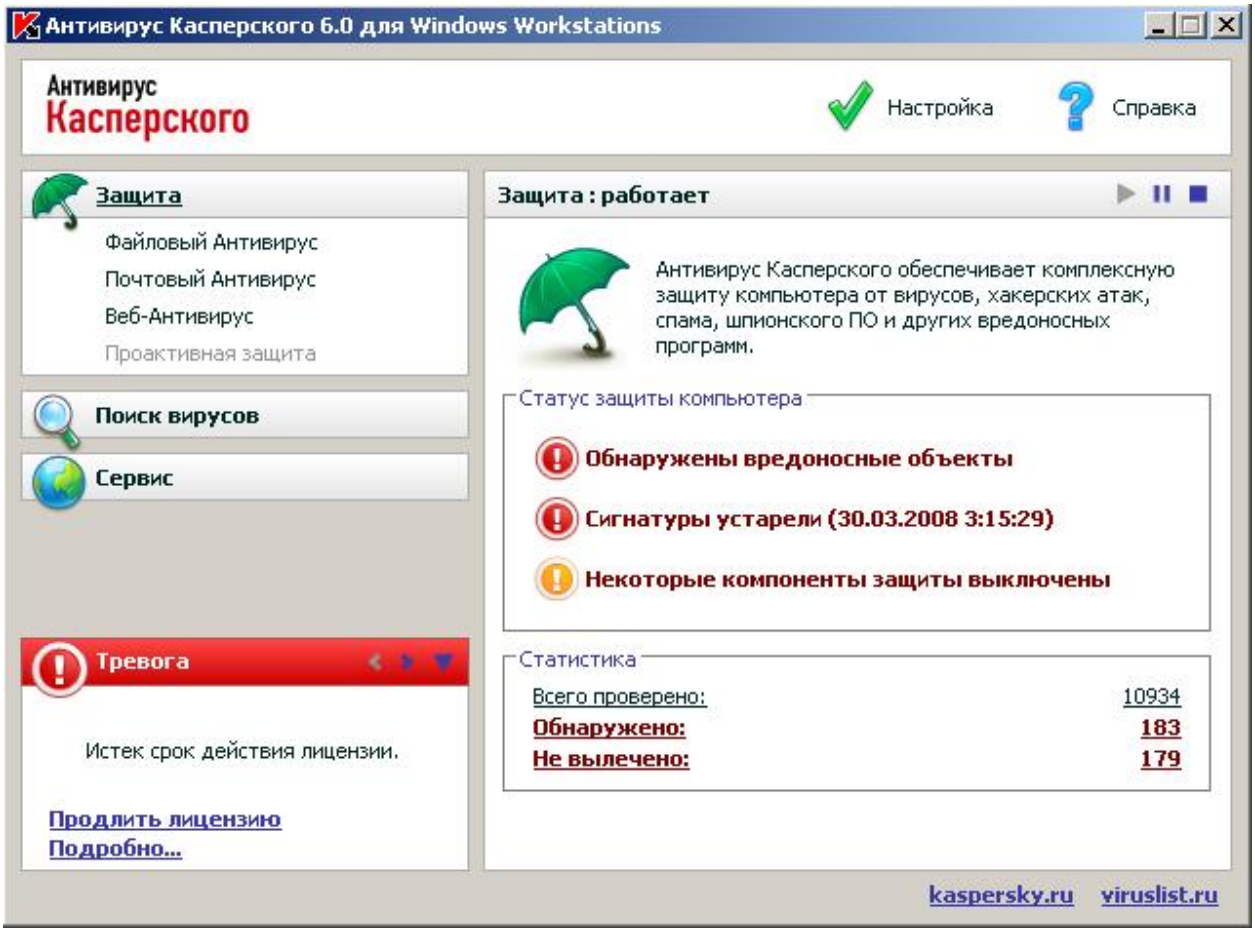


Рис. 11. Выбор контейнера «Статистика» в главном окне программы

В окне «Защита» нажать кнопку «Сохранить как» и сохранить отчет в текстовый файл

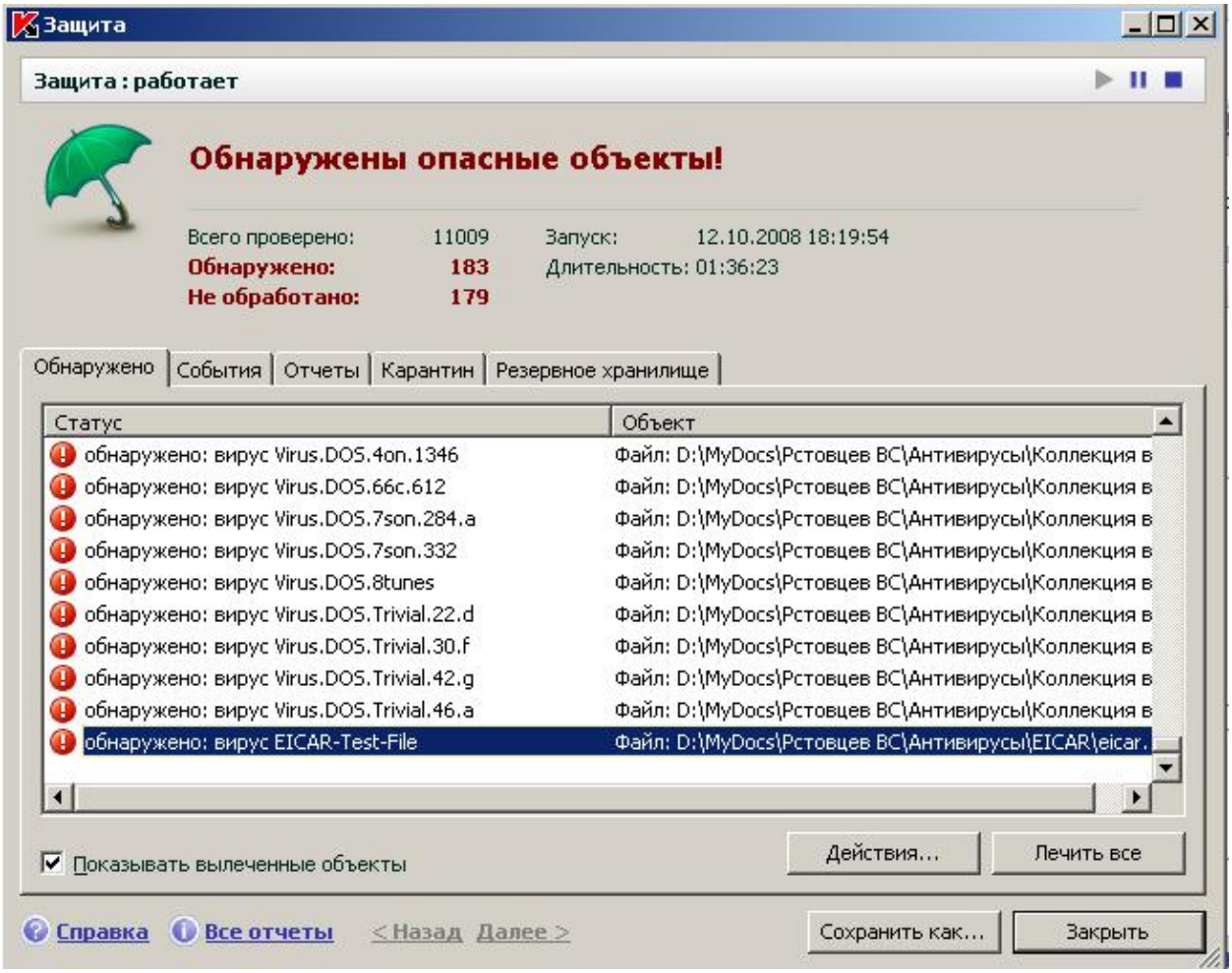


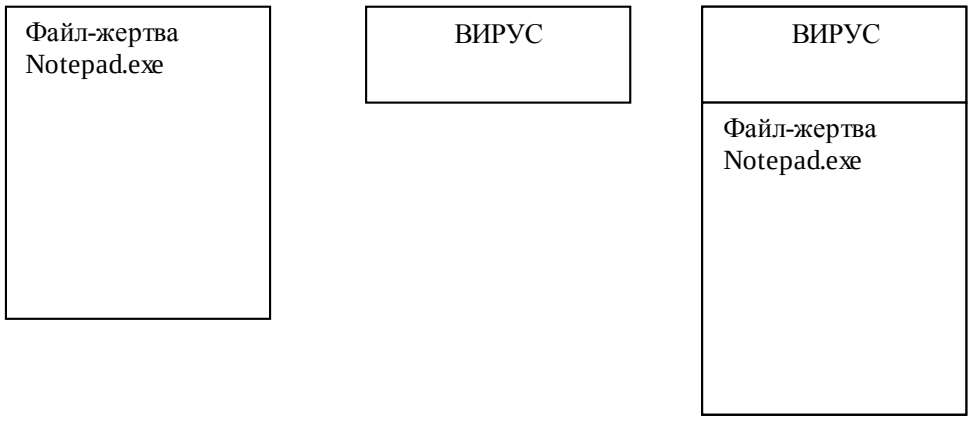
Рис. 12. Статистика обнаружения вирусов.

1.4. Изучение механизма работы файловых вирусов

Файловые вирусы записывают свое тело в файл-жертву и используют различные механизмы получения управления с целью выполнения деструктивных функций и дальнейшего распространения. Один из способов заключается в добавлении тела вируса в начало файла-жертвы.

Изначально вирус существует сам по себе. Пусть тем или иным образом он был запущен на выполнение.

1. Вирус ищет файл-жертву.
2. Переименовывает его, задавая ему случайное имя.
3. Создает новый файл в том же каталоге с исходным именем файла-жертвы.
4. Открывает для чтения свой исполняемый файл
5. Записывает в файл, созданный на шаге 3 свой код
6. Открывает для чтения файл-жертву, переименованный на шаге 2
7. Дозаписывает в файл, созданный на шаге 3 код файла-жертвы.
8. Удаляет исходный файл жертвы, переименованный на шаге 2



При запуске инфицированного файла-жертвы вирус выполняется первым и выполняет следующие действия:

1. Проверяет размер запущенного файла в байтах.
2. Если размер больше размера самого вируса, значит запущен инфицированный файл. Иначе – запущен вирус в исходной форме.
3. Вирус создает файл с двойным расширением *.exe.exe (Notepad.exe.exe) в том же каталоге.
4. Смещается в инфицированном файле в место начала настоящей программы Notepad.exe
5. Считывает код программы Notepad.exe и записывает его в файл Notepad.exe.exe
6. Запускает на выполнение файл Notepad.exe.exe
7. Выполняет полезную нагрузку (вредоносные действия)
8. Выполняет поиск новых файлов-жертв и их заражение.

Литература

1 Скрипник Д.А. Общие вопросы технической защиты информации. — Электрон. текст. дан. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — Режим доступа : http ://www.iprbookshop.ru/52161.— ЭБС «IPRbooks», по паролю.

2 Лапонина О.Р. Основы сетевой безопасности. Криптографические алгоритмы и протоколы взаимодействия. — Электрон. текст. дан.— М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — Режим доступа : http ://www.iprbookshop.ru/52217.— ЭБС «IPRbooks», по паролю.

3 Алексеев, А.П.; Многоуровневая защита информации Электронный ресурс : монография / А.П. Алексеев. - Самара : Поволжский государственный университет телекоммуникаций и информатики, 2017. - 128 с. - Книга находится в базовой версии ЭБС IPRbooks. - ISBN 978-5-904029-72-2

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ

НЕВИННОМЫССКИЙ ТЕХНОЛОГИЧЕСКИЙ ИНСТИТУТ (ФИЛИАЛ)

Методи ческие указания к самостоятельным работам
по дисциплине
"Информационная безопасность изащита данных"

Направление подготовки 15.04.04
«Автоматизация технологических процессов и производств»
Направленность (профиль) «Информационно-управляющие системы»
Форма обучения - очно-заочная
Год начала обучения 2022
Реализуется в 5 семестре

Невинномысск 2022

Методические указания предназначены для студентов направления 15.04.04 Автоматизация технологических процессов и производств и других технических специальностей. Они содержат рекомендации по организации самостоятельных ра-бот студента на правления 15.04.04 для дисциплины «Защита информации в системах управления».

Методические указания разработаны в соответствии с требованиями ФГОС ВО в части содержания и уровня подготовки выпускников направления 15.04.04 Автоматизация технологических процессов и производств.

Наименование компетенций:

Код	Формулировка
ПК-9	Способность обеспечивать надежность и безопасность на всех этапах жизненного цикла продукции, выбирать системы экологической безопасности производства

Составитель канд. техн. наук Кочеров Ю. Н.

Ответственный редактор канд. техн. наук Д.В. Болдырев

Содержание

1 Подготовка к лабораторным занятиям

Для того чтобы лабораторные занятия приносили максимальную пользу, 4
необходимо помнить, что упражнение и решение задач проводятся по рассмотренному на лекциях материалу и связаны, как правило, с детальным разбором отдельных вопросов лекционного курса. Следует подчеркнуть, что только после усвоения лекционного материала с определенной точки зрения (а именно с той, с которой он излагается на лекциях) он будет закрепляться студентом на лабораторных занятиях как в результате обсуждения и анализа лекционного материала, так и с помощью решения проблемных ситуаций, задач. При этих условиях студент не только хорошо усвоит материал, но и научится применять его на практике, а также получит дополнительный стимул (и это очень важно) для активной проработки лекции.

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками.

Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

2 Подготовка к практическим занятиям

Подготовку к каждому практическому занятию студент должен начать с ознакомления с методическими указаниями, которые включают содержание работы. Тщательное продумывание и изучение вопросов основывается на проработке текущего материала лекции, а затем изучения обязательной и дополнительной литерату-

ры, рекомендованную к данной теме. На основе индивидуальных предпочтений студенту необходимо самостоятельно выбрать тему доклада по проблеме и по возможности подготовить по нему презентацию.

Если программой дисциплины предусмотрено выполнение практического задания, то его необходимо выполнить с учетом предложенной инструкции (устно или письменно). Все новые понятия по изучаемой теме необходимо выучить наизусть и внести в глоссарий, который целесообразно вести с самого начала изучения курса. Результат такой работы должен проявиться в способности студента свободно ответить на теоретические вопросы семинара, его выступлении и участии в коллективном обсуждении вопросов изучаемой темы, правильном выполнении практических заданий и контрольных работ.

В зависимости от содержания и количества отведенного времени на изучение каждой темы практическое занятие может состоять из четырех-пяти частей:

1. Обсуждение теоретических вопросов, определенных программой дисциплины.

2. Доклад и/ или выступление с презентациями по выбранной проблеме.

3. Обсуждение выступлений по теме – дискуссия.

4. Выполнение практического задания с последующим разбором полученных результатов или обсуждение практического задания.

5. Подведение итогов занятия.

Первая часть – обсуждение теоретических вопросов – проводится в виде фронтальной беседы со всей группой и включает выборочную проверку преподавателем теоретических знаний студентов. Примерная продолжительность — до 15 минут. Вторая часть — выступление студентов с докладами, которые должны сопровождаться презентациями с целью усиления наглядности восприятия, по одному из вопросов практического занятия. Обязательный элемент доклада – представление и анализ статистических данных, обоснование социальных последствий любого экономического факта, явления или процесса. Примерная продолжительность — 20-25 минут. После докладов следует их обсуждение – дискуссия. В ходе этого этапа практического занятия могут быть заданы уточняющие вопросы к докладчикам. Примерная продолжительность – до 15-20 минут. Если программой предусмотрено

выполнение практического задания в рамках конкретной темы, то преподавателями определяется его содержание и дается время на его выполнение, а замет идет обсуждение результатов. Подведением итогов заканчивается практическое занятие.

В процессе подготовки к практическим занятиям, студентам необходимо обратить особое внимание на самостоятельное изучение рекомендованной учебно-методической (а также научной и популярной) литературы. Самостоятельная работа с учебниками, учебными пособиями, научной, справочной и популярной литературой, материалами периодических изданий и Интернета, статистическими данными является наиболее эффективным методом получения знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у студентов свое отношение к конкретной проблеме. Более глубокому раскрытию вопросов способствует знакомство с дополнительной литературой, рекомендованной преподавателем по каждой теме семинарского или практического занятия, что позволяет студентам проявить свою индивидуальность в рамках выступления на данных занятиях, выявить широкий спектр мнений по изучаемой проблеме.

3 Самостоятельное изучение темы. Конспект

Конспект – наиболее совершенная и наиболее сложная форма записи. Слово «конспект» происходит от латинского «conspectus», что означает «обзор, изложение». В правильно составленном конспекте обычно выделено самое основное в изучаемом тексте, сосредоточено внимание на наиболее существенном, в кратких и четких формулировках обобщены важные теоретические положения.

Конспект представляет собой относительно подробное, последовательное изложение содержания прочитанного. На первых порах целесообразно в записях ближе держаться тексту, прибегая зачастую к прямому цитированию автора. В дальнейшем, по мере выработки навыков конспектирования, записи будут носить более свободный и сжатый характер.

Конспект книги обычно ведется в тетради. В самом начале конспекта указывается фамилия автора, полное название произведения, издательство, год и место издания. При цитировании обязательная ссылка на страницу книги. Если цитата взята

из собрания сочинений, то необходимо указать соответствующий том. Следует помнить, что четкая ссылка на источник – непременное правило конспектирования. Если конспектируется статья, то указывается, где и когда она была напечатана.

Конспект подразделяется на части в соответсвтии с заранее продуманным планом. Пункты плана записываются в тексте или на полях конспекта. Писать его рекомендуется четко и разборчиво, так как небрежная запись с течением времени становиться малопонятной для ее автора. Существует правило: конспект, составленный для себя, должен быть по возможности написан так, чтобы его легко прочитал и кто-либо другой.

Формы конспекта могут быть разными и зависят от его целевого назначения (изучение материала в целом или под определенным углом зрения, подготовка к докладу, выступлению на занятии и т.д.), а также от характера произведения (монография, статья, документ и т.п.). Если речь идет просто об изложении содержания работы, текст конспекта может быть сплошным, с выделением особо важных положений подчеркиванием или различными значками.

В случае, когда не ограничиваются переложением содержания, а фиксируют в конспекте и свои собственные суждения по данному вопросу или дополняют конспект соответствующими материалами их других источников, следует отводить место для такого рода записей. Рекомендуется разделить страницы тетради пополам по вертикали и в левой части вести конспект произведения, а в правой свои дополнительные записи, совмещая их по содержанию.

Конспектирование в большей мере, чем другие виды записей, помогает вырабатывать навыки правильного изложения в письменной форме важные теоретических и практических вопросов, умение четко их формулировать и ясно излагать своими словами.

Таким образом, составление конспекта требует вдумчивой работы, затраты времени и труда. Зато во время конспектирования приобретаются знания, создается фонд записей.

Конспект может быть текстуальным или тематическим. В текстуальном конспекте сохраняется логика и структура изучаемого произведения, а запись ведется в

соответствии с расположением материала в книге. За основу тематического конспекта берется не план произведения, а содержание какой-либо темы или про-

блемы.

Текстуальный конспект желательно начинать после того, как вся книга прочитана и продумана, но это, к сожалению, не всегда возможно. В первую очередь необходимо составить план произведения письменно или мысленно, поскольку в соответствии с этим планом строится дальнейшая работа. Конспект включает в себя тезисы, которые составляют его основу. Но, в отличие от тезисов, конспект содержит краткую запись не только выводов, но и доказательств, вплоть до фактического материала. Иначе говоря, конспект – это расширенные тезисы, дополненные рассуждениями и доказательствами, мыслями и соображениями составителя записи.

Как правило, конспект включает в себя и выписки, но в него могут войти отдельные места, цитируемые дословно, а также факты, примеры, цифры, таблицы и схемы, взятые из книги. Следует помнить, что работа над конспектом только тогда будет творческой, когда она не ограничена текстом изучаемого произведения. Нужно дополнять конспект данными из другими источников.

В конспекте необходимо выделять отдельные места текста в зависимости от их значимости. Можно пользоваться различными способами: подчеркиваниями, вопросительными и восклицательными знаками, репликами, краткими оценками, писать на полях своих конспектов слова: «важно», «очень важно», «верно», «характерно».

В конспект могут помещаться диаграммы, схемы, таблицы, которые придадут ему наглядность.

Составлению тематического конспекта предшествует тщательное изучение всей литературы, подобранной для раскрытия данной темы. Бывает, что какая-либо тема рассматривается в нескольких главах или в разных местах книги. А в конспекте весь материал, относящийся к теме, будет сосредоточен в одном месте. В плане конспекта рекомендуется делать пометки, к каким источникам (вплоть до страницы) придется обратиться для раскрытия вопросов. Тематический конспект составляется обычно для того, чтобы глубже изучить определенный вопрос, подготовиться к докладу, лекции или выступлению на семинарском занятии. Такой конспект по содер-

жанию приближается к реферату, докладу по избранной теме, особенно если включает и собственный вклад в изучение проблемы.

9

4 Комплект заданий для выполнения контрольной работы

Каждый студен применяет пороговую (k,n) схему разделения данных Миньот-10 та к своей Фамилии имени и отчеству (основания схемы должны отличаться от схемы представленной в примере).

Алгорит разделения данных с применением схемы Миньотта:

Пороговая схема разделения данных Миньотта использует специальные последовательности целых чисел, называемые последовательностями Миньотта.

Пусть n целое число, $n \geq 2$, и $2 \leq k \leq n$. Где (k,n) – это последовательность Миньотта. Числа Миньотта должны быть попарно взаимно простыми числами и должны соблюдаться условия: $p_1 < p_2 < ... < p_n$ и $\prod_{i=0}^{k-2} p_{n-1} < \prod_{i=1}^{k} p_n$.

При использовании (n,k) последовательности Миньотта, пороговая схема функционирует следующим образом:

– информация S выбирается как случайное целое число, такое что: $\beta < M < \alpha$, где $\alpha = \prod_{i=1}^{k} p_i$ и $\beta = \prod_{i=0}^{k-2} p_{n-1}$, т.е. информация должна находиться в диапазоне $(p_1 \cdot p_2 \cdot ... \cdot p_k; p_{n-k+2} \cdot p_{n-k+3} \cdot ... p_n)$;

– часть данных I_i рассчитывается как $I_i = S \bmod p_i$, для всех $1 \leq i \leq n$;

– получаем k различных частей $I_{i_1},...,I_{i_k}$, информация S восстанавливается с помощью $p_{i_1} ... p_{i_k}$ системы:

$$\begin{cases} x \equiv I_{i_1} \bmod p_{i_1} \\ ... \\ x \equiv I_{i_k} \bmod p_{i_k} \end{cases} \qquad (1)$$

Пример:

Расспотрим (k,n) разделения данных k=3, n=5.

Возьмер набор простых чисел p_1=2, p_2=3, p_3=5, p_4=7, p_5=11. Тогда из $\beta < M < \alpha$ видно что информация находится в диапазоне 30<M<77.

Возьмем M=50 тогда:

x_1=M *mod* p_1=55 *mod* 2=1;

x_2=M *mod* p_2=55 *mod* 3=1;

x_3=M *mod* p_3=55 *mod* 5=0

x_4=M *mod* p_4=55 *mod* 7=6

x_5=M *mod* p_5=55 *mod* 11=0

востановление информации возможно с помощью Китайской теореммы об остатках (КТО), обобщенной полиадической системы счисления (ОПСС) или совместного применения КТО и ОПСС.

Ниже представлен алгоритм преобразования из системы остаточных классов в позиционную систему счисления на основе Китайской Теоремы об Остатках

11

Метод нахождения числа по его остаткам был найден в Китае две тысячи лет назад. Основой этого метода является теорема, названная Китайской теоремой об остатках.

Пусть $p_1, p_2, p_3,..., p_n$ попарно взаимно простые числа. $P = \prod_{i=1}^{n} p_i$, $m_1, m_2, m_3,..., m_n$ подобраны так, что $\frac{P}{p_i} m_i \equiv 1 (\mathrm{mod}\, p_i)$, а $A_0 = \sum_{i=1}^{n} \frac{P}{p_i} \cdot m_i \cdot \alpha_i$ где $i = \overline{1,n}$. Тогда решение системы $A \equiv \alpha_i (\mathrm{mod}\, p_i)$, $i = \overline{1,n}$ будет иметь вид $A \equiv A_0 (\mathrm{mod}\, P)$.

Эта теорема лежит в основе метода ортогональных базисов при переводе из СОК в ПСС.

Пусть $p_1, p_2, p_3,..., p_n$ – это основания СОК. Тогда $P = \prod_{i=1}^{n} p_i$ – это объем диапазона системы. С выбором системы определяются ее основные константы – базисы B_i, $i = \overline{1,n}$. Задача перевода числа $A = (\alpha_1, \alpha_2, \alpha_3,.., \alpha_n)$ в ПСС заключается в определении чисел M_i, $i = \overline{1,n}$, чтобы $A = \sum_{i=1}^{n} M_i B_i$. Для однозначного определения M_i на базисы B_i накладывается ряд ограничений и показывается, что свойством обладают базисы $B_1 = (1,0,0,...,0)$, $B_2 = (0,1,0,...,0)$, $B_3 = (0,0,1,...,0)$ … $B_n = (0,0,0,...,1)$, которые называются ортогональными.

Тогда, в случае ортогональности базисов $M_i = \alpha_i$, $i = \overline{1,n}$, и базисы определяются по формуле $B_i = \frac{P}{p_i} m_i = P_i m_i$, $i = \overline{1,n}$, где $P_i = \frac{P}{p_i}$.

m_i – это числа, называемые весами базисов, и их получают из сравнений $P_i m_i \equiv 1 (\mathrm{mod}\, P_i)$.

Тогда по КТО число $A = (\alpha_1, \alpha_2, \alpha_3,.., \alpha_n) = \sum_{i=1}^{n} B_i \alpha_i (\mathrm{mod}\, P)$.

Отсюда следует, что если найдены ортогональные базисы, то для перевода числа $A = (\alpha_1, \alpha_2, \alpha_3,.., \alpha_n)$ в ПСС достаточно вычислить $\sum_{i=1}^{n} B_i \alpha_i$ и ввести в диапазон

$[0;P)$ вычитанием величины кратной P, т.е. $\left|\sum_{i=1}^{n} B_i\alpha_i\right|_P = \sum_{i=1}^{n} B_i\alpha_i - r_A P$, где r_A – ранг числа A, показывающий, сколько раз нужно вычесть величину диапазона P, чтобы вернуть его в диапазон.

Т.к. B_i определяется выбранным основанием, она может быть вычислена заранее единственный раз.

Восстановление информации методом, основанным на КТО, сводится к вычислению $A = (B_1 \cdot \alpha_1 + B_2 \cdot \alpha_2 + B_3 \cdot \alpha_3 + ... + B_n \cdot \alpha_n) \bmod P$.

Рассмотрим привер восстановленя информации для трех частей данных
$x_1 = 1$, $x_3 = 0$, $x_5 = 0$ по основания $p_1=2$, $p_3=5$, $p_5=11$.
Тогда $P=p_1 * p_3 * p_5=2*5*11=110$
Расчитаем коэффичиенты:

$$P_1 = \frac{P}{p_1} = \frac{110}{2} = 55$$
$$P_3 = \frac{P}{p_3} = \frac{110}{5} = 22$$
$$P_5 = \frac{P}{p_5} = \frac{110}{11} = 10$$

Далее расчитает веса базисов из приближеня:

$$P_i \cdot m_i mod p_i \equiv 1$$
$$55 \cdot m_1 mod 2 \equiv 1 \text{ тогда } m_1 = 1$$
$$22 \cdot m_3 mod 5 \equiv 1 \text{ тогда } m_1 = 3$$
$$10 \cdot m_5 mod 11 \equiv 1 \text{ тогда } m_1 = 10$$

Вычислим веса базисов:

$$B_i = P_i \cdot m_i$$
$$B_1 = 55 \cdot 1 = 55$$
$$B_3 = P_3 \cdot m_3 = 66$$
$$B_i = P_i \cdot m_i = 100$$

Востановим информацию применив формулу
$A = (B_1 \cdot \alpha_1 + B_2 \cdot \alpha_2 + B_3 \cdot \alpha_3 + ... + B_n \cdot \alpha_n) \bmod P$

$$M = (1 \cdot 55 + 0 \cdot 66 + 0 \cdot 110) mod 110 = 55\ mod 110 = 55$$

Литература

1 Скрипник Д.А. Общие вопросы технической защиты информации. — Электрон. текст. дан. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — Режим доступа : http ://www.iprbookshop.ru/52161.— ЭБС «IPRbooks», по паролю. 13

2 Лапонина О.Р. Основы сетевой безопасности. Криптографические алгоритмы и протоколы взаимодействия. — Электрон. текст. дан.— М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — Режим доступа : http ://www.iprbookshop.ru/52217.— ЭБС «IPRbooks», по паролю.

3 Алексеев, А.П.; Многоуровневая защита информации Электронный ресурс : монография / А.П. Алексеев. - Самара : Поволжский государственный университет телекоммуникаций и информатики, 2017. - 128 с. - Книга находится в базовой версии ЭБС IPRbooks. - ISBN 978-5-904029-72-2