МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

Государственное образовательное учреждение
высшего профессионального образования
"Оренбургский государственный университет"

Кафедра вычислительной техники

Т.Н. ШАЛКИНА

МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ КОМПЬЮТЕРНОЙ ИНФОРМАЦИИ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
К ЛАБОРАТОРНОМУ ПРАКТИКУМУ

Рекомендовано к изданию Редакционно-издательским советом
государственного образовательного учреждения высшего профессионального
образования "Оренбургский государственный университет"

Оренбург 2006

УДК 004(07)
ББК 32.81я7
Ш 18

Рецензент
доктор технических наук, профессор В.Н. Тарасов

Шалкина Т.Н.
Ш 18 **Методы и средства защиты компьютерной информации: методические указания к лабораторному практикуму/ Т.Н. Шалкина - Оренбург, ГОУ ОГУ, 2006. – 44 с.**

Методические указания предназначены для проведения лабораторных работ по общепрофессиональной дисциплине «Методы и средства защиты компьютерной информации» для студентов специальностей направления "Информатика и вычислительная техника.

ББК 32.81я7

Содержание

1 Лабораторная работа № 1. Основы теории чисел

1.1 Постановка задачи

Цель работы: изучить основные понятия, теоремы и алгоритмы теории чисел, используемые в криптографии.

Задание: Составить схему алгоритма и написать программу, реализующую следующие функции:

–нахождения наибольшего общего делителя двух чисел на основе алгоритма Евклида;

–нахождения последовательности простых чисел, не превосходящих данного N, на основе алгоритма Эратосфена;

–отыскания функции Эйлера для положительного целого числа;

–отыскания решения линейного сравнения.

Полученные алгоритмы оформить в виде отдельного модуля (библиотеки).

1.2 Теоретические предпосылки

Основа любого криптографического алгоритма базируется на элементах теории чисел, изучению которых и посвящена данная лабораторная работа. Теория чисел занимается изучением свойств целых чисел. Сразу оговоримся, что при изложении теоретического материала мы будем обозначать буквами только целые числа.

1. Основные понятия.

Определение 1. Если a делится на b нацело, мы будем говорить, что b делит a. При этом a называется кратным числа b, b – делителем числа a. Число a можно представить как

$$a = q \cdot b ,$$

где q – полное частное.

Теорема 1 Если в равенстве вида $k + l + ... + n = p + q + ... + s$ относительно всех членов, кроме какого-либо одного, известно, что они кратны b, то и этот один член кратен b.

Доказательство:

Пусть таким одним членом будет k. Имеем

$l = b \cdot l_1$

$n = b \cdot n_1$

$p = b \cdot p_1$

$q = b \cdot q_1$

$s = b \cdot s_1$

Перенесем в правую часть равенства все члены, кроме k.

$k = p + q + ... + s - l - ... - n = b \cdot (p_1 + q_1 + ... + s_1 - l_1 - ... - n_1)$

Таким образом, k представляется произведением b на целое число ($p_1 + q_1 + ... + s_1 - l_1 - ... - n_1$) и тем самым делится на b по определению 1. Что и требовалось доказать.

Теорема 2 (о делении с остатком).

Всякое целое a представляется единственным способом с помощью положительного целого b равенством вида

$$a = b \cdot q + r; \qquad 0 \le r \prec b \tag{1}$$

Доказательство

Действительно, одно представление числа a равенством вида (1) получим, взяв $b \cdot q$ равным наибольшему кратному числа b, не превосходящему a.

Допустим существование другого представления числа a еще одним равенством вида (1)

$$a = b \cdot q_1 + r_1; \qquad 0 \le r_1 \prec b \tag{2}$$

Вычтем почленно равенство (2) из (1), получим

$$0 = b \cdot (q - q_1) + r - r_1 \tag{3}$$

Согласно Теореме 1 разность $r - r_1$ кратна b. С другой стороны разность двух неотрицательных чисел меньших b сама будет численно меньше b. Числом кратным b и численно меньшим b является 0. Если $r - r_1 = 0$, то из равенства (3) следует, что и $q - q_1 = 0$. Таким образом, второе представление числа a тождественно первому. Теорема 2 доказана.

Число q называется *неполным частным,* а число r – *остатком от деления* a на b.

2. Наибольший общий делитель.

Определение 2.Число, которое делит каждое из чисел a и b, называется общим делителем чисел a и b.

Определение 3. Наибольший из делителей чисел a и b называется наибольшим общим делителем (НОД) этой пары чисел.

Обозначение. НОД (a,b) ≡ (a,b).

Определение 4. Если НОД(a,b)=1, то числа a и b называются попарно

простыми.

Теорема 3. Если a кратно b, то совокупность общих делителей чисел a и b совпадает с совокупностью делителей одного числа b; в частности (a,b)=b.

Доказательство:

Действительно, всякий общий делитель чисел a и b является делителем и одного b. Раз a кратно b, то, согласно Теореме 1, всякий делитель числа b является также делителем числа a, т.е. является общим делителем чисел a и b.

Таким образом, совокупность общих делителей чисел a и b совпадает с совокупностью делителей одного b. Поскольку наибольший делитель числа b есть само b, то (a,b)=b.

Теорема 4. Если $a = b \cdot q + r$ (2), то совокупность общих делителей чисел a и b совпадает с совокупностью делителей чисел b и c; в частности (a,b)=(b,c).

Доказательство:

Действительно, всякий общий делитель чисел a и b согласно Теореме 3 является и общим делителем чисел b и c. Всякий общий делитель чисел b и c делит a и, следовательно, является общим делителем чисел a и b.

Таким образом, общие делители чисел a и b те же, что и общие делители чисел b и c. В частности, должны совпадать и наибольшие из этих делителей, (a,b)=(b,c).

Алгоритм Евклида. Применяется для отыскания НОД двух чисел.

Пусть a,b – положительные целые и a>b.

Согласно Теореме 3 находим ряд равенств:

$$\begin{aligned}
&a = b \cdot q_1 + r_2, && 0 \le r_2 \prec b,\\
&b = r_2 \cdot q_2 + r_3, && 0 \le r_3 \prec r_2,\\
&r_2 = r_3 \cdot q_3 + r_4, && 0 \le r_4 \prec r_3,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&r_{n-2} = r_{n-1} \cdot q_{n-1} + r_n, && 0 \le r_n \prec r_{n-1},\\
&r_{n-1} = r_n \cdot q_n,
\end{aligned} \qquad (4)$$

заканчивающийся, когда получается некоторое $r_{n+1}=0$. Последнее неизбежно, так как ряд $b,r_2,r_3,\ldots$ не может содержать более чем b положительных чисел.

Таким образом, НОД чисел a и b равен последнему не равному нулю остатку алгоритма Евклида. Поскольку, согласно Теореме 4

$$(a,b) = (b,r_2) = (r_2,r_3) = \ldots = (r_{n-1},r_n) = r_n.$$

Пример. Отыскать НОД(25,9) - ?

$$25 = 9*2 + 7$$
$$9 = 7*1 + 2$$
$$7 = 2*3 + 1$$
$$2 = 1*2 + 0$$
$$НОД(25,9) = 1$$

3. Простые числа.

Определение 5. Всякое a>1 будем называть простым, если у него нет других делителей, кроме 1 и самого себя, иначе – составным.

Определение 6. Два числа называются попарно простыми, если их НОД равен 1.

Алгоритм Эратосфена построения последовательности простых чисел в ряду целых чисел, не превосходящих данного целого N.

Выписываем ряд чисел

$$1,2,\ldots, N \qquad (5)$$

Первое простое число в ряду (5) – 2. Вычеркиваем из ряда (5) все числа кратные 2, кроме самого числа 2. Первое, оставшееся после 2, простое число – 3. Вычеркиваем из ряда (5) все числа кратные 3, кроме самого числа 3. Первое, следующее за 3, невычеркнутое простое число 5. Вычеркиваем из ряда (5) все числа кратные 5, кроме числа 5. И т.д.

Когда указанным способом вычеркнуты все числа, кратные простых, меньше простого p, то все невычеркнутые меньшие p^2 будут простые.

Выводы (без доказательства):

1) приступая к вычеркиванию кратных простого p, это вычеркивание следует начать с p^2;

2) составление последовательности простых чисел, не превосходящих N, закончено, как только вычеркнуты все составные кратные простых, не превосходящих $\sqrt{N}$.

Теорема 5. Всякое целое, большее единицы, разлагается на произведение простых сомножителей и притом единственным способом:

$$a = p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdot \ldots \cdot p_K^{\alpha_K}, \qquad (6)$$

где $p_1,\ldots,p_K$ – простые сомножители числа a,

$\alpha_1,\ldots,\alpha_K$ – кратности вхождения соответственно сомножителей $p_1,\ldots,p_K$ в число a.

Разложение (6) числа a на простые сомножители называется *канониче-*

ским.

Пример. $18 = 2^1 \cdot 3^2$

Определение 7. Функцией Эйлера $\varphi(a)$ называется функция, которая для $\forall a \in Z_+$, равна количеству чисел в ряду от 1 до a-1 попарно простых с a, где $a \geq 1$.

Пример.

$\varphi(1) = 1$	$\varphi(4) = 2$	$(1,3)$
$\varphi(2) = 1$	$\varphi(5) = 4$	$(1,2,3,4)$
$\varphi(3) = 2$	$\varphi(6) = 2$	$(1,5)$ и тт.д

Теорема 6. Пусть $a = p_1^{\alpha_1} \cdot p_2^{\alpha_2} \cdot \ldots \cdot p_k^{\alpha_k}$ - каноническое разложение числа a. Тогда имеем

$$\varphi(a) = a \cdot \left(1 - \frac{1}{p_1}\right) \cdot \left(1 - \frac{1}{p_2}\right) \cdot \ldots \cdot \left(1 - \frac{1}{p_k}\right)$$

или также

$$\varphi(a) = \left(p_1^{\alpha_1} - p_1^{\alpha_1 - 1}\right) \cdot \left(p_2^{\alpha_2} - p_2^{\alpha_2 - 1}\right) \cdot \ldots \cdot \left(p_k^{\alpha_k} - p_k^{\alpha_k - 1}\right).$$

(без доказательства).

В частности, если p – простое, то $\varphi(p) = p - 1$.

4. Вычеты. Линейные сравнения.

Определение 8. Пусть m – некоторое целое положительное число m>1. Пусть a и b – это числа, которые при делении на m имеют один и тот же остаток:

$$a = m \cdot t_1 + r \qquad 0 \leq r < m$$
$$b = m \cdot t_2 + r$$

Числа a и b будем называть равноостаточными.

Очевидно, таких чисел бесчисленное множество, т.е. они образуют класс чисел равноостаточных по модулю m или, как говорят, сравнимых по модулю m.

Обозначение. $a = b \bmod m$.

Пример. $7 = 10 \bmod 3 \qquad 7 = 10 + 3 \cdot (-1)$.

Все сравнимые числа отличаются на кратное число модулей и поэтому могут быть представлены в виде $a = b + m \cdot t$.

Сравнение первой степени, или линейное сравнение

$$a \cdot x + b = 0 \bmod m \qquad (7)$$

можно представить в виде:

$$a \cdot x = b \bmod m \qquad (8)$$

перенесением свободного члена в правую часть.

Среди множества различных линейных сравнений, нас будут интересовать только те, для которых выполняется следующее условие (a,b)=1, поскольку в этом случае линейное сравнение будет иметь решение и притом единственное.

Рассмотрим алгоритм решения линейного сравнения, основанный на алгоритме Евклида.

1. С помощью алгоритма Евклида находят вектор неполных частных $q_1, \ldots, q_n$ для чисел m и a (порядок чисел в данном случае принципиален).

2. Делают обозначение: $P_0=1$, $P_1=q_1$.

3. По формуле

$$P_i = q_i \cdot P_{i-1} + P_{i-2}$$

находят значения вектора P_i для i=2..n-1.

4. Решение линейного сравнения находят по формуле

$$x = (-1)^{n-1} \cdot P_{n-1} \cdot b \bmod m$$

Пример. Решить линейное сравнение $7x = 5 \bmod 19$

Решение

1. Находим неполные частные с использованием алгоритма Евклида для чисел m и a.

$$19 = 7 \cdot 2 + 5 \quad q_1 = 2$$
$$7 = 5 \cdot 1 + 2 \quad q_2 = 1$$
$$5 = 2 \cdot 2 + 1 \quad q_3 = 2$$
$$2 = 1 \cdot 2 \quad q_4 = 2$$

n=4

2. $P_0=1$, $P_1=q_1$.

3.

$$P_2 = q_2 \cdot P_1 + P_0 \qquad P_2 = 1 \cdot 2 + 1 = 3$$
$$P_3 = q_3 \cdot P_2 + P_1 \qquad P_3 = 2 \cdot 3 + 2 = 8$$

4. $x = (-1)^3 \cdot P_3 \cdot b \bmod m = (-1) \cdot 8 \cdot 5 \bmod 19 = -40 \bmod 19 = -2 \bmod 19 = 17 \bmod 19$

1.3 Упражнения

1. Найдите НОД(396,1452)?
2. Найти каноническое разложение числа а)2156; б)1934; в)1132.
3. Являются ли числа попарно простыми

а) (678, 941); б) (243, 1485); в) (535, 321)?

4. Решить линейное сравнение:
а) $5x = 4 \bmod 21$; б) $11x = 17 \bmod 119$; в) $12x = 8 \bmod 67$

1.4 Вопросы к лабораторной работе № 1

1. Докажите теорему о делении с остатком.
2. Дайте понятие НОД двух чисел? Какой алгоритм используется для его нахождения?
3. Какие числа называются простыми, составными?
4. Расскажите алгоритм нахождения последовательности простых чисел.
5. Какое разложение целого числа называется каноническим?
6. Что показывает функция Эйлера, как она рассчитывается?
7. Расскажите алгоритм решения линейных сравнений.

2 Лабораторная работа № 2. Криптографические системы

2.1 Постановка задачи

Цель работы: изучить основные криптографические алгоритмы.

Задание: Составить схему алгоритма и написать программу, реализующую:

–симметричных алгоритмов шифрования простой и сложной заменой;
–один из асимметричных алгоритмов шифрования (по выбору).

2.2 Теоретические предпосылки

Криптография представляет собой совокупность методов преобразования данных, направленных на то, чтобы сделать эти данные бесполезными для противника. Иначе, криптография обеспечивает сокрытие смысла сообщения с помощью шифрования и открытие его расшифровыванием, которые выполняются по специальным криптографическим алгоритмам с помощью ключевой информации.

В общем виде работу любой криптографической системы можно представить следующим образом. Отправитель генерирует открытый текст исходного сообщения M, которое должно быть передано законному получателю по незащищенному каналу. За каналом следит перехватчик с целью перехватить и раскрыть передаваемое сообщение. Для того, чтобы перехватчик не смог узнать содержание сообщения, отправитель шифрует его с помощью обратимого преобразования, тем самым получая шифртекст сообщения $C = E_к(M)$, который затем отправляет получателю.

Получатель, приняв шифртекст C, расшифровывает его с помощью обратного преобразования $D=E_к^{-1}$ и получает исходное сообщение M.

$$D_k(C)=E_k^{-1}(E_k(M))=M$$

Преобразование $E_к$ выбирается из семейства криптографических преобразований, называемых криптоалгоритмами (шифрами). Параметр, с помощью которого выбирается отдельное используемое преобразование, называется криптографическим ключом.

Преобразование шифрования может быть симметричным или ассиметричным относительно преобразования расшифрования. Это важное свойство функции преобразования определяет два класса криптосистем:

- симметричные (одноключевые) криптосистемы;
- ассиметричные (двухключевые) криптосистемы (или криптосистемы с открытым ключом).

Основной характеристикой шифра является криптостойкость, которая определяет его стойкость к раскрытию методами криптоанализа. Обычно эта характеристика определяется интервалом времени, необходимым для раскрытия шифра.

1. Симметричные криптосистемы.

В общем виде схему криптографической симметричной системы шифрования данных можно представить в следующем виде (рисунок 1).

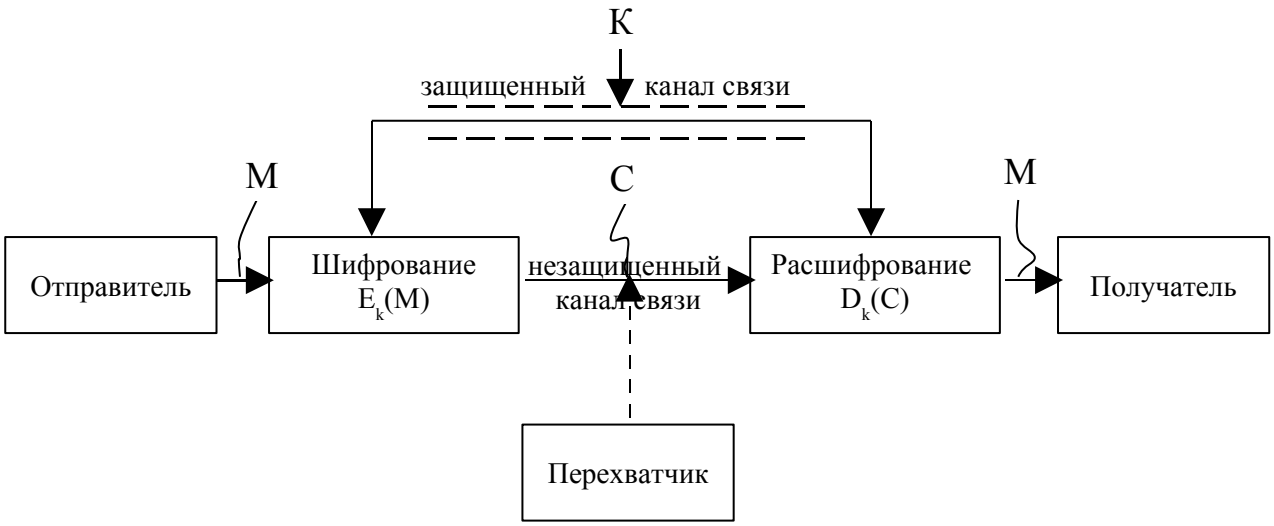

Рисунок 1

Характерной особенность симметричных систем шифрования является то, что для шифрования и расшифрования сообщений применяется один и тот же ключ, который вследствие этого является секретным и должен передаваться получателю по секретным каналам связи.

Симметричные алгоритмы шифрования можно сгруппировать следующим образом:

- шифры перестановок;
- шифры замены;
- шифры гаммирования;
- шифры, основанные на аналитических преобразованиях шифруемых данных.

Шифрование перестановкой заключается в том, что символы шифруемого текста переставляются по определенному правилу в пределах некоторого блока текста. При достаточной длине блока, в пределах которого осуществляется перестановка, и сложном неповторяющемся порядке перестановки можно достигнуть приемлемой для простых практических приложений стойкости шифра.

Шифрование заменой (подстановкой) заключается в том, что символы шифруемого текста заменяются символами того же или другого алфавита в соответствии с заранее обусловленной схемой замены.

Шифрование гаммированием заключается в том, что символы шифруемого текста складываются с символами некоторой случайной последовательности, именуемой гаммой шифра.

Шифрование аналитическим преобразованием заключается в том, что шифруемый текст преобразуется по некоторому аналитическому правилу (формуле).

Как открытый текст, так и шифртекст образуются из букв, входящих в конечное множество символов, называемых алфавитом (например, множество всех заглавных и строчных букв, множество всех заглавных букв и т.п.). При выполнении криптографических преобразований полезно заменить буквы алфавита целыми числами, например для русского алфавита, А - 0, Б - 1, С - 2 и т.д.

Более подробно рассмотрим машинную реализацию шифров простой и сложной замены. В настоящее время эти методы шифрования уже не используются самостоятельно, но очень часто встречаются как составные элементы других более сложных криптографических преобразований, например, как DES или ГОСТ 28147-89.

Шифры простой замены.

При шифровании заменой (подстановкой) символы шифруемого текста заменяются символами того же или другого алфавита с заранее установленным правилом замены. Часто шифры простой замены называют шифрами одноалфавитной подстановки.

Многим, наверное, известен шифр замены, связанный с именем Юлия Цезаря. В процессе шифрования каждая буква исходного текста заменялась четвертой по счету от нее в алфавите: A-B-C-D, или D вместо A. Послание сенату VENI VIDI VICI, то есть ПРИШЕЛ УВИДЕЛ ПОБЕДИЛ, сделанное Цезарем после однодневной войны с понтийским царем Фарнаком, выглядело бы шифровкой SBKF SFAF SFZF.

Для компьютерной реализации, процесс шифрования простой замены может быть описан формулой:

$$y_i = (x_i + k) \bmod m, \qquad (9)$$

где n - количество букв в сообщении;
m - количество букв в алфавите сообщения;
i=1..n; 0 <=k<m.

Процесс расшифрования:

$$x_i = (y_i - k) \bmod m \qquad (10)$$

Алгоритм шифрования методом простой замены можно описать следующим образом:

1) каждому символу алфавита сообщения проставляется в соответствие цифровое обозначение;

2) символы исходного сообщения и ключа заменяются на цифровое значение в соответствии с п.1;

3) по формуле (9) рассчитываются цифровые значения символов шифртекста;

4) полученные значения в соответствии с п.1 переводятся в символьное обозначение.

Алгоритм рашифрования аналогичен данному, отличие заключается в формуле, используемой в п. 3, для расшифрования следует использовать формулу (10).

Пример. Зашифруем сообщение ЮСТАС АЛЕКСУ, используя к=Б.

1. Преобразуем символы алфавита к цифровому обозначению (таблица 1).

Таблица 1 - Русский алфавит

_	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32

2. Проставим соответствие (таблица 2).

Таблица 2

Ю	С	Т	А	С	-	А	Л	Е	К	С	У
31	18	19	1	18	0	1	12	6	11	18	20

Ключ к=2.

3. Подставляя данные в формулу (9), получим числовые значения символов шифртекста:

$31 + 2 = 33 \bmod 33 = 0$
$18 + 2 = 20 \bmod 33 = 20$
$19 + 2 = 21 \bmod 33 = 21$
$1 + 2 = 3 \bmod 33 = 3$
$18 + 2 = 20 \bmod 33 = 20$
$0 + 2 = 2 \bmod 33 = 2$
$1 + 2 = 3 \bmod 33 = 3$
$12 + 2 = 14 \bmod 33 = 14$
$6 + 2 = 8 \bmod 33 = 8$
$11 + 2 = 13 \bmod 33 = 13$
$18 + 2 = 20 \bmod 33 = 20$
$20 + 2 = 22 \bmod 33 = 22$

4. Заменив, полученные числовые значение на символы, получим шифр-

текст:

_УФВУБВНЗМУХ

Шифр сложной замены

Шифры сложной замены называются многоалфавитными (в отличие от одноалфавитных - шифров простой замены), так как для шифрования каждого символа исходного сообщения применяют свой шифр замены.

Эффект использования многоалфавитной подстановки заключается в том, что обеспечивается маскировка естественной статистики исходного языка, поскольку конкретный символ исходного текста может преобразован в разные символы шифртекста.

Многоалфавитные шифры замены предложил и ввел в практику Леон Батист Альберти, который также был известным теоретиком искусства и архитектором. Его книга "Трактат о шифре", написанная в 1566 г, представляла собой первый в Европе научный труд по криптологии. Поэтому криптологи всего мира считают Л. Альберти основоположником криптологии.

Однако наиболее известной из многоалфавитных систем является система Вижинера (по имени французского дипломата XVI века Блеза Вижинера).

Система шифрования Вижинера в современной трактовке, применимой для реализации на ЭВМ, преобразует открытый текст $x=(x_0, \ldots, x_{n-1})$ в шифртекст $y=(y_0, \ldots, y_{n-1})$ с помощью ключа $k=(k_0, \ldots, k_t)$ согласно следующему правилу:

$$y_i = (x_i + k_j) \bmod m, \qquad (11)$$

где n - количество букв в сообщении;
m - количество символов в алфавите;
t – количество символов в ключе;
i=0…n-1;
j= i mod t.

Процесс расшифрования можно представить в виде:

$$x_i = (y_i - k_j) \bmod m, \qquad (12)$$

Таким образом, можно сказать, что на каждом шаге система шифрования Вижинера представляет собой простую (одноалфавитную) замену. Отличие заключается в том, что ключ меняется от буквы к букве.

Пример. Зашифруем сообщение ПРИХОДИТЕ ЗАВТРА с ключом КОТ.

1. Согласно таблице 1, представим исходное сообщение в числовом виде (таблица 3).

Таблица 3

П	Р	И	Х	О	Д	И	Т	Е	_	З	А	В	Т	Р	А
16	17	10	22	15	5	10	19	6	0	8	1	3	19	17	1

Представим ключ также в цифровой форме (таблица 4).

Таблица 4

К	О	Т
11	15	19

2. Получим шифртекст согласно по формуле (11).

16 + 11 = 27 mod 33 = 27
17 + 15 = 32 mod 33 = 32
10 + 19 = 29 mod 33 = 29
22 + 11 = 33 mod 33 = 0
15 + 15 = 30 mod 33 = 30
5 + 19 = 24 mod 33 = 24
10 + 11 = 21 mod 33= 21
19 + 15 = 34 mod 33 = 1
6 + 19 = 25 mod 33 = 25
0 + 11 = 11 mod 33 = 11
8 + 15 = 23 mod 33 = 23
1 + 19= 20 mod 33 = 20
3 + 11= 14 mod 33 = 14
19 + 15 = 34 mod 33 = 1
17 + 19 = 36 mod 33 = 3
1 + 11 = 12 mod 33 = 12

3. Шифртекст в символьной форме: ЪЯЬ_ЭЧФАШКЦУНАВЛ

Таким образом, символ "Р", встречающаяся дважды в исходном тексте в шифртексте встречается как "Я" и как "В" (это касается и других встречающихся несколько раз в исходном тексте символов). Может возникнуть ситуация, когда различные в исходном тексте символы будут зашифрованы одинаково.

Рассмотренные криптосистемы имеют свои достоинства и недостатки шифра сложной замены.К достоинствам можно отнести простоту реализации и высокую скорость работы алгоритма.

К недостаткам:

- не маскируются статистические свойства текста;
- сохраняется алфавитный порядок в последовательности заменяющих букв;
- число возможных значений ключей мало;
- легко вскрываются на основе анализа частот появления букв в шифртексте или перебора всех возможных ключей.

2. Асимметричные криптосистемы.

Общую схему функционирования асимметричной криптосистемы можно представить следующим образом (рисунок 2).

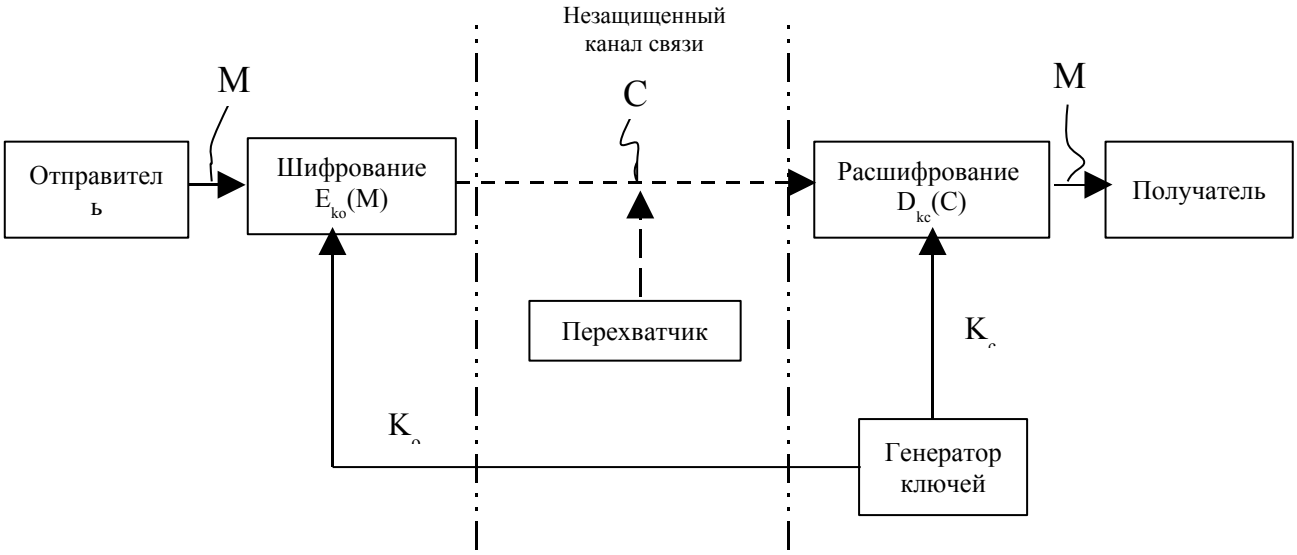

Рисунок 2

В традиционных криптосистемах одним и тем же секретным ключом осуществляется как шифрование, так и расшифрование сообщения. Это предполагает, что отправитель и получатель сообщения получили идентичные копии ключа курьером. Этот прием почти неприменим для коммерческих фирм и абсолютно недоступен частным лицам из-за своей дороговизны.

При шифровании с открытым ключом для шифрования и расшифровывания используются разные ключи, и знание одного из них не дает практической возможности определить второй. Поэтому ключ для шифрования может быть сделан общедоступным без потери стойкости шифра, если ключ для расшифровывания сохраняется в секрете, например, генерируется и хранится только получателем информации.

Защита информации в асимметричной системе основана на секретности закрытого ключа.

У. Диффи и М. Хеллман сформулировали требования, выполнение которых обеспечит безопасность асимметричной криптосистемы:

1) вычисление пары ключей (открытого и закрытого) должно быть простым;

2) отправитель, зная открытый ключ k_o легко вычисляет криптограмму

$$C=E_{ko}(M);$$

3) получатель, используя закрытый ключ k_c и криптограмму C, легко восстанавливает исходное сообщение:

$$M=D_{kc}(C);$$

4) противник, зная открытый ключ, при попытке вычислить секретный ключ наталкивается на непреодолимую вычислительную проблему;

5) противник, зная открытый ключ и криптограмму, при попытке восстановить исходное сообщение наталкивается на непреодолимую вычислительную проблему.

Однонаправленные функции.

Концепция асимметричных криптосистем основана на применении однонаправленных функций. Неформально можно дать следующее определение однонаправленной функции. *Однонаправленной* называется функция $F : X \rightarrow Y$, обладающая двумя свойствами:

а) существует полиномиальный алгоритм вычисления значений F(x);

б) не существует полиномиального алгоритма инвертирования функции F (т.е. решения уравнения F(x) = y относительно x).

В криптографии широко применяются следующие однонаправленные функции.

Разложение большого числа (используется в криптосистеме RSA) на простые сомножители:

$$N = P \cdot Q$$

По современным оценкам теории чисел при целом $N \approx 2^{664}$ и приблизительно равных сомножителях потребуется около 10^{23} операций, т.е. эта задача практически неразрешима на современной ЭВМ.

Задача дискретного логарифмирования (например, используется в криптосистеме El Gamal):

$$A^x \bmod N = y$$

Алгоритм вычисления дискретного логарифма пока не найден, по этому данная функция пока считается однонаправленной (однако не доказано, что эффективного алгоритма не существует). При целых $A \approx 2^{664}$ и $N \approx 2^{664}$ решение задачи дискретного логарифмирования (нахождение показателя степени x для известного y) потребуется около 10^{26} операций, т.е. эта задача имеет еще большую вычислительную сложность, чем предыдущая.

Криптосистема RSA.

Первая и наиболее известная криптографическая система с открытым ключом была предложенна в 1978 году и получила название RSA. Ее название происходит от первых букв фамилий авторов Rivest, Shamir и Adleman. Она основана на трудности разложения очень больших целых чисел на простые сомножители.

Процедуру шифрования данных алгоритмом RSA (как и любым другим асимметричным алгоритмом) можно разбить на два этапа: определение пары ключей (секретного и открытого) и непосредственно шифрование данных.

I Определение ключей.

1. Получатель выбирает два очень больших простых числа P и Q и вычисляет их произведение

$$N = P \cdot Q$$

2. Получатель вычисляет функцию Эйлера от модуля N по формуле

$$m = \varphi(N) = (P - 1) \cdot (Q - 1)$$

4. Затем он выбирает случайное целое число K_o, исходя из следующих условий

5.

$$1 < K_o \le \varphi(N) \quad \text{и} \quad (K_o, \varphi(N)) = 1$$

K_o и N объявляются открытыми ключами и пересылаются отправителю.

4. Закрытый ключ K_c находится из решения линейного сравнения

$$K_o \cdot K_c = 1 \bmod \varphi(N)$$

II Шифрование/расшифрование данных.

1. Если M – сообщение (длина которого, определяемая по значению выражаемого им целого числа, должна принадлежать интервалу (1, N)), то отправитель вычисляет криптограмму по следующей формуле

$$C = M^{Ko} \bmod N$$

2. Получатель сообщения расшифровывает его следующим образом

$$M = C^{Kc} \bmod N$$

Пример. Зашифровать сообщение M=15, используя следующие входные данные: P=23, Q=37, K_o=47, а затем расшифровать полученную криптограмму.

I Определяем ключи.

1. Вычисляем модуль N

$$N = P \cdot Q = 23 \cdot 37 = 851$$

2. Рассчитываем функцию Эйлера от модуля N

$$\varphi(N) = (P - 1) \cdot (Q - 1) = (23 - 1) \cdot (37 - 1) = 22 \cdot 36 = 792$$

3. Значения K_o=47 и N=851 объявляются открытыми ключами и пересылаются отправителю.

4. Находим закрытый ключ K_c, решив следующее линейное

$$47 \cdot K_c = 1 \bmod 792$$

а) находим неполные частные с использованием алгоритма Евклида для чисел 792 и 19.

$$792 = 47 \cdot 16 + 40 \qquad q_1 = 16$$
$$47 = 40 \cdot 1 + 7 \qquad q_2 = 1$$
$$40 = 7 \cdot 5 + 5 \qquad q_3 = 5$$
$$7 = 5 \cdot 1 + 2 \qquad q_4 = 1$$
$$5 = 2 \cdot 2 + 1 \qquad q_5 = 2$$
$$2 = 1 \cdot 2 \qquad q_6 = 2$$

n=6

б) $P_0=1$, $P_1=q_1$.

в)

$$P_2 = q_2 \cdot P_1 + P_0 \qquad P_2 = 1 \cdot 16 + 1 = 17$$
$$P_3 = q_3 \cdot P_2 + P_1 \qquad P_3 = 5 \cdot 17 + 16 = 101$$
$$P_4 = q_4 \cdot P_3 + P_2 \qquad P_4 = 1 \cdot 101 + 17 = 118$$
$$P_5 = q_5 \cdot P_4 + P_3 \qquad P_5 = 2 \cdot 118 + 101 = 337$$

г)

$$K_c = (-1)^5 \cdot P_5 \cdot b \bmod m = (-1) \cdot 337 \bmod 792 = 455 \bmod 792$$

II Шифруем сообщение.

1. M=15 (M<N)

$$C = M^{Ko} \bmod N = 15^{47} \bmod 851 = 500$$

3. Проверим, расшифровав сообщение

$$M = C^{Kc} \bmod N = 500^{455} \bmod 851 = 15$$

Криптосистема El Gamal (Эль Гамаля).

Еще одной широко известной асимметричной криптосистемой является система, которую предложил в 1985 году Эль Гамаль.

I Определение ключей.

1. С целью генерации пары ключей вначале выбирается большое простое число P и большое целое число G. Эти значения являются открытыми.

2. Выбирается случайное число X (X<P), которое является секретным ключом.

3. Вычисляется

$$Y = G^X \bmod P$$

Y объявляется открытым ключом.

II Шифрование/расшифрование данных.

Сообщения представляются целыми числами M из интервала (1, P).

1. Для шифрования сообщения M (длина которого (1, P)), выбирается случайное число K, 1<K<P-1, (K,P-1)=1.

2. Отправитель вычисляет следующие числа

$$a = G^K \bmod P$$
$$b = Y^K \cdot M \bmod P$$

Пара чисел (a,b) является криптограммой исходного сообщения M.

3. Получатель расшифровывает криптограмму следующим образом:

$$M = b / a^X \bmod P = b \cdot a^{-X} \bmod P$$

Доказательство:

$$b / a^X = Y^K \cdot M / a^X = G^{XK} \cdot M / G^{KX} \bmod P = M \bmod P$$

Пример. Зашифровать сообщение M=40, используя следующие входные данные: P=157, G=83, X=34 (секретный ключ), а затем расшифровать полученную криптограмму.

I Определение ключей.

1. Значения P и G, которые являются открытыми, уже даны в условии задачи, так же как и X.

2. Вычислим открытый ключ

$$Y = G^X \bmod P = 83^{34} \bmod 157 = 25$$

II Шифрование данных.

1. Исходя из следующих условий, 1<K<P-1, (K,P-1)=1, выберем K=29.

2. Вычисляем криптограмму

$$a = G^K \bmod P = 83^{29} \bmod 157 = 84$$
$$b = Y^K \cdot M \bmod P = 25^{29} \cdot 40 = 110$$

Пара чисел (84,110) отправляется получателю.

3. Расшифруем криптограмму:

$$M = b / a^X \bmod P = b \cdot a^{-X} \bmod P$$

Для расшифрования полученного сообщения предварительно найдем значение $a^{-x} \bmod P$, решив линейное сравнение

$$a^X \cdot a^{-X} = 1 \bmod P$$

Для удобства сделаем следующее обозначение

$$y = a^{-X}$$

Следовательно, наше линейное сравнение будет выглядеть следующим образом

$$84^{34} \cdot y = 1 \bmod 157$$

$$42 \cdot y = 1 \bmod 157$$

а) находим неполные частные с использованием алгоритма Евклида для чисел 157 и 42.

$$157 = 42 \cdot 3 + 31 \qquad q_1 = 3$$
$$42 = 31 \cdot 1 + 11 \qquad q_2 = 1$$
$$31 = 11 \cdot 2 + 9 \qquad q_3 = 2$$
$$11 = 9 \cdot 1 + 2 \qquad q_4 = 1$$
$$9 = 2 \cdot 4 + 1 \qquad q_5 = 4$$
$$2 = 1 \cdot 2 \qquad q_6 = 2$$

n=6

б) $P_0=1$, $P_1=q_1$

в)

$$P_2 = q_2 \cdot P_1 + P_0 \qquad P_2 = 1 \cdot 3 + 1 = 4$$
$$P_3 = q_3 \cdot P_2 + P_1 \qquad P_3 = 2 \cdot 4 + 3 = 11$$
$$P_4 = q_4 \cdot P_3 + P_2 \qquad P_4 = 1 \cdot 11 + 4 = 15$$
$$P_5 = q_5 \cdot P_4 + P_3 \qquad P_5 = 4 \cdot 15 + 11 = 71$$

г)

$$y = (-1)^5 \cdot P_5 \cdot b \bmod P = (-1) \cdot 71 \bmod 157 = 86 \bmod 157$$

Используя найденное значение, рассчитываем исходное сообщение M

$$M = 110 \cdot 86 \bmod 157 = 40$$

2.3 Вопросы к лабораторной работе № 2

1. Что изучает наука криптография?
2. Что понимается под шифром, ключом?
3. Расскажите общий принцип действия любой криптографической системы.
4. Особенности функционирования симметричных криптосистем. На чем основана криптостойкость симметричных алгоритмов шифрования?
5. Перечислите и кратко охарактеризуйте основные виды симметричных алгоритмов шифрования.
6. Расскажите суть шифров простой и сложной замены, их основные недостатки.
7. Какие современные симметричные криптосистемы Вы знаете?
8. Особенности функционирования асимметричных алгоритмов шифрования.
9. Дайте понятие односторонней функции. Какие односторонние функции используются в криптографических алгоритмах шифрования?

10. Расскажите основные асимметричные криптографические алгоритмы RSA и El Gamal, на чем основана их криптостойкость?

3 Лабораторная работа № 3. Изучение службы Active Directory операционной системы Windows 2000

3.1 Постановка задачи

1 Изучить структуру и назначение Active Directory и ее объектов.

2 Научиться управлять группами, компьютерами, учетными записями пользователей.

3.2 Теоретические предпосылки

Структура и назначение Active Directory и ее объектов.

Служба каталогов Active Directory является средством для именования, хранения и выборки информации в некоторой распределенной среде, доступное для приложений, пользователей и различных клиентов этой среды. Служба сетевых каталогов хранит информацию об общедоступных приложениях, файлах, принтерах и сведения о пользователях.

Служба каталогов Active Directory обеспечивает эффективную работу сложной корпоративной среды, предоставляя следующие возможности.

Единая регистрация в сет. Пользователи могут регистрироваться в сети *с одним* именем и паролем и получать при этом доступ ко всем сетевым ресурсам (серверам, принтерам, приложениям, файлам и т. д.) независимо от их расположения в сети.

Безопасность информации. Средства аутентификации и управления доступом к ресурсам, встроенные в службу Active Directory, обеспечивают централизованную защиту сети. Права доступа можно определять не только для каждого *объекта* каталога, но и каждого *свойства* (атрибута) объекта.

Централизованное управление. Администраторы могут централизованно управлять всеми корпоративными ресурсами.

Администрирование с использованием групповых политик. При загрузке компьютера или регистрации пользователя в системе выполняются требования групповых политик; их настройки хранятся в объектах групповых политик (GPO) и "привязываются" к сайтам, доменам или организационным единицам. Групповые политики определяют, например, права доступа к различным объектам каталога или ресурсам, а также множество других правил работы в системе.

Интеграция с DNS. Служба Active Directory тесно связана с DNS. Этим достигается единство в именовании ресурсов локальной сети и сети Интернет, в результате чего упрощается подключение пользовательской сети к Интернету.

Масштабируемость. Служба Active Directory может охватывать как один домен, так и множество доменов, один контроллер домена или множество контроллеров домена, т. е. она отвечает требованиям сетей любого масштаба. Несколько доменов можно объединить в дерево доменов, а несколько деревьев доменов можно связать в лес.

Стандартные интерфейсы. Для разработчиков приложений служба каталогов предоставляют доступ ко всем возможностям (средствам) каталога и поддерживают принятые стандарты и интерфейсы программирования (API). Служба каталогов тесно связана с операционной системой что позволяет избежать дублирования в прикладных программах функциональных возможностей системы, например, средств безопасности.

Рассмотрим основные понятия, используемые в Active Directory.

Каталог состоит из *элементов* (entries), представляющих собой информацию, или *атрибуты,* связанные с некоторым реальным *объектом,* например компьютером, человеком или организацией.

Каждый объект принадлежит хотя бы к одному *объектному классу,* представляющему собой некоторое семейство объектов с определенными общими характеристиками. Класс объектов определяет тип информации, содержащейся в Active Directory для экземпляров (объектов) данного класса. Атрибуты могут быть как *обязательными* (mandatory) для данного класса (например, имя), так и *дополнительными* (optional) (пароль).

Контейнер (container) - это специфический объект службы каталогов, который, в отличие от обычных объектов, не имеет какого-либо физического представления, а служит только структурной организации других объектов каталога. Типичным примером контейнеров могут служить *организационные единицы,* или *подразделения,* используемые для упрощения администрирования отдельных групп ресурсов или пользователей в домене.

Элементы каталога организованы в виде *иерархического дерева,* называемого Directory Information Tree (DIT, Информационное дерево каталога или просто Дерево каталога). Элементы, находящиеся ближе к корню дерева, обычно представляют крупные объекты, например, организации или компании; элементы, располагающиеся на ветвях этого дерева (листья) представляют более простые объекты - пользователей, устройства, компьютеры.

Схема каталога (Directory Schema) - это набор правил, описывающих структуру дерева каталога, объявления и синтаксис объектных классов и типы атрибутов, входящих в каталог.

Схема каталога гарантирует, что все добавления или изменения каталога соответствуют данным правилам, и препятствует появлению некорректных элементов, ошибочных типов атрибутов или классов.

В Active Directory схема реализована как набор экземпляров объектных классов, хранящийся в самом каталоге. Этим Active Directory отличается от многих каталогов, в которых схема хранится в текстовом файле, считываемом при запуске каталога. Когда схема хранится в каталоге, пользовательские приложения могут обращаться к ней и узнавать об имеющихся объектах и свойствах. Схему Active Directory можно динамически обновлять: модифицировать и расширять.

Основные компоненты любой службы каталога - база данных, содержащая нужную информацию, и один или несколько протоколов, обеспечивающих

доставку данных пользователям.

Active Directory обеспечивает хранение любой общедоступной информации. Как и другие службы каталогов, Active Directory обеспечивает некоторый механизм хранения информации и протоколы для доступа к ней.

Компьютеры на базе Windows 2000 объединяются в домены. *Домены* - это известное решение для администрирования групп, предоставляющее каждому пользователю учетную запись в конкретном домене. Однако, в отличие от Windows NT Server 4.0, где доменам давались простые строковые имена (имена NetBIOS), в среде Windows 2000 Server каждый домен должен иметь имя, отвечающее соглашениям именования доменов Domain Name System (DNS). В каждом домене один или несколько компьютеров должны выполнять функции контроллеров домена.

В среде Windows 2000 Server каждый контроллер домена содержит полную копию базы данных Active Directory этого домена. В Active Directory используются так называемое ядро Extended Storage Engine (ESE) и два различных протокола, обеспечивающих связь между клиентами и базой данных. Для поиска контроллера домена клиент обращается к протоколу, описанному в DNS. Для доступа к данным в Active Directory клиент использует протокол Lightweight Directory Access Protocol (LDAP).

В большинстве современных сетей TCP/IP используется служба DNS, главное назначение которой преобразовывать символьные имена в IP-адреса. Для этого каждый компьютер-сервер DNS имеет набор записей с информацией о ресурсах. Каждая запись имеет некоторый тип, определяющий характер и назначение хранящейся информации. Интеграцию служб Active Directory и DNS можно рассматривать в трех аспектах:

–домены Active Directory и домены DNS имеют одинаковую иерархическую структуру и схожее пространство имен;

–зоны (zone) DNS могут храниться в Active Directory. Если используется сервер DNS, входящий в состав Windows 2000 Server, то первичные зоны (primary zone), занесенные в каталог, реплицируются на все контроллеры домена, что обеспечивает лучшую защищенность службы DNS.

Каждый элемент Active Directory и каждый атрибут любого элемента имеют список управления доступом (ACL), который определяет права и возможности пользователей в отношении доступа к конкретным элементам и атрибутам. Например, список ACL может позволить одним пользователям читать атрибуты некоторого элемента, другим пользователям – читать и изменять некоторые из атрибутов, а остальным – запретить какой-либо доступ к элементу. Эффективное управление доступом невозможно без достоверной аутентификации клиентов, Active Directory использует для этой цели протокол Kerberos.

Управление подразделениями, компьютерами, группами и учетными записями пользователей.

Для управления учетными записями пользователей и компьютерами следует вначале войти в раздел администрирования (Administrative Tools) и выбрать Active Directory Users and Computers (рисунок 3).

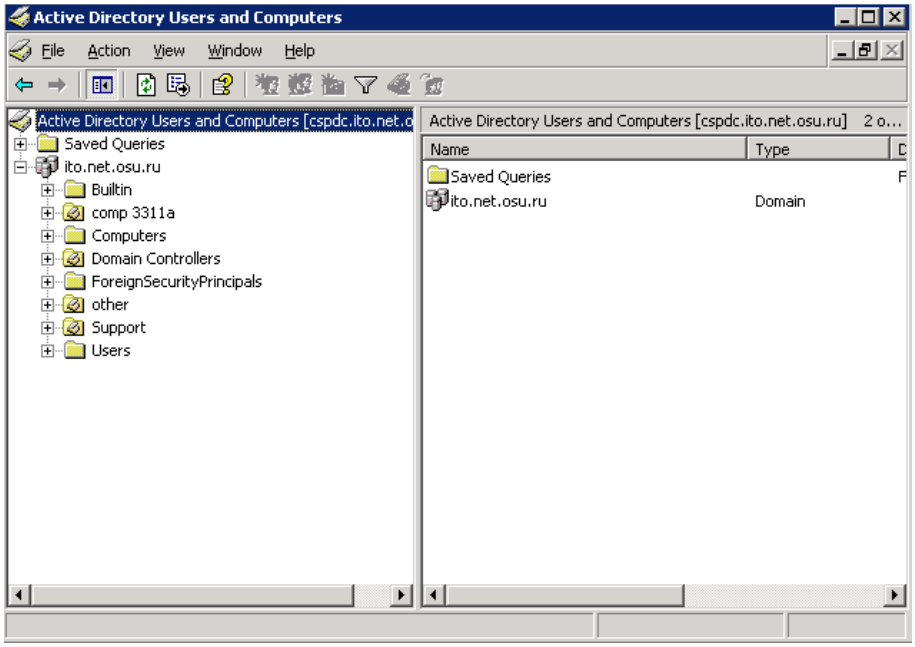

Рисунок 3

Создание подразделения (организационной единицы).

Для создания *подразделения,* или *организационной единицы* (Organizational Unit, OU) следует:

1. Выделите объект типа "домен" (на рисунке 1 домен имеет название ito.net.osu.ru) и нажмите правую кнопку мыши. В появившемся меню выберите команду **Создать | Подразделение** (New | Organizational Unit). Можно воспользоваться панелью инструментов и кнопкой **Создание нового подразделения в текущем контейнере** (Create a new organizational unit in a current container) на панели инструментов.

2. В открывшемся окне укажите имя создаваемого подразделения и нажмите кнопку **ОК.**

В результате в выбранном вами домене будет создано подразделение с заданным именем. В дальнейшем внутри него можно создать вложенные подразделения.

На рисунке 1 представлено несколько подразделений – comp3311a, other, Support, Domain Controllers.

Создание группы.

В процессе установки домена Windows 2000 в нем создается несколько встроенных групп, обладающих определенным набором прав. Их можно использовать для присвоения администраторам или пользователям определенных ролей или прав доступа в домене.

К встроенным относятся перечисленные ниже группы. Эти группы служат для назначения разрешений доступа пользователям, на которых возложено выполнение в данном домене каких-либо административных функций.

Локальные группы в домене:

- Администраторы (Administrators)
- Гости (Guests)
- Операторы архива (Backup Operators)
- Операторы печати (Print Operators)
- Операторы сервера (Server Operators)
- Операторы учета (Account Operators)
- Пользователи (Users)
- Репликатор (Replicator)
- Совместимый с пред-Windows 2000 доступ (Pre-Windows 2000 CompatibleAccess)

Глобальные группы:

- Администраторы домена (Domain Admins)
- Владельцы-создатели групповой политики (Group Policy Creator Owners)
- Гости домена (Domain Guests)
- Издатели сертификатов (Cert Publishers)
- Компьютеры домена (Domain Computers)
- Контроллеры домена (Domain Controllers)
- Пользователи домена (Domain Users)

Универсальные группы:

- Администраторы предприятия (Enterprise Admins)
- Администраторы схемы (Schema Admins)

Универсальные группы создаются только на контроллерах корневого (первого в лесе) домена. В зависимости от установленных на сервере служб могут быть и дополнительные встроенные группы, локальные в домене или глобальные. По умолчанию все встроенные локальные группы домена находятся в папке **Builtin** объекта домена (рисунок 2). Все встроенные глобальные группы находятся в папке **Users.** Встроенные группы можно переносить в другие контейнеры или подразделения в пределах домена.

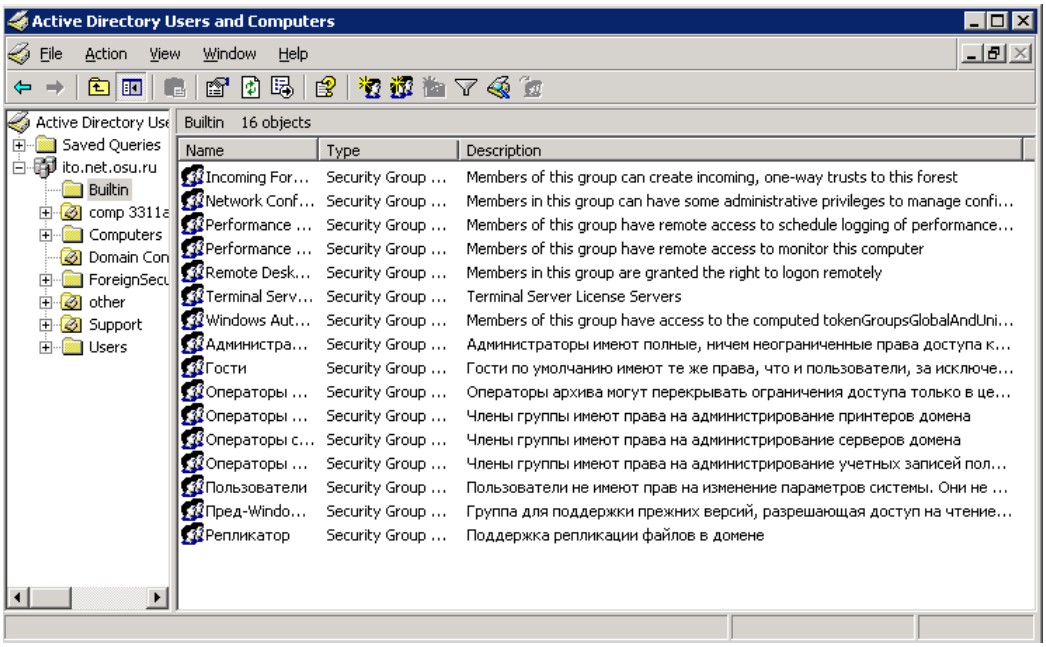

Рисунок 2

По умолчанию каждая созданная в домене учетная запись автоматически становится членом группы **Пользователи домена**. Кроме того, группа **Пользователи домена** является членом локальной в домене группы **Пользователи**.

Любой объект типа **Компьютер** (Computer) при создании по умолчанию автоматически включается в группу **Компьютеры домена.**

Группа **Администраторы домена** объединяет всех пользователей, имеющих полный административный доступ в домене. По умолчанию **Администраторы домена** являются членами локальной в домене группы **Администраторы**.

Группа **Гости домена** объединяет все учетные записи, с помощью которых можно зарегистрироваться в домене без пароля и получить минимальные права доступа. По умолчанию **Гости домена** являются членами локальной в домене группы **Гости**.

Помимо перечисленных выше встроенных групп администратор может создать любое количество групп пользователей и предоставить им необходимый набор прав и разрешений. Для создания группы необходимо выполнить следующее:

1. Выберите подразделение, где следует создать группу, и нажмите правую кнопку мыши. Выберите в появившемся меню команду **Создать** | **Группа** (Group), либо нажмите кнопку **Создание новой группы в текущем контейнере** (Create New Group in a Current Container) на панели инструментов.

2. В открывшемся окне диалога **Новый объект – Группа (New Object - Group)** в поле **Имя группы** (Group name) введите имя создаваемой группы.

3. Установите переключатель **Тип группы** (Group type) в одно из положений, соответствующее типу создаваемой группы: **Группа безопасности** (Security) или **Группа распространения** (Distribution). Первый тип группы служит для предоставления пользователям определенного набора прав доступа к

таким ресурсам сети, как файлы и принтеры. Второй тип группы служит только для распространения информации в сети, например в качестве списков рассылки электронной почты. Следует отметить, что группы безопасности могут использоваться в качестве групп распространения.

4. Установив в одно из положений переключатель **Область действия группы** (Group scope), выберите подходящую область действия создаваемой группы. *Область действия группы* определяет, где может быть видна данная группа *(уровень доступности)* и какие типы объектов могут быть ее членами, и может быть выбрана как:

–локальная в домене (Domain Local): пользователи, а также глобальные и универсальные группы из всего леса, другие локальные группы из этого же домена;

–глобальная (Global): пользователи, а также глобальные и универсальные группы;

–универсальная (Universal): пользователи и глобальные группы (только в основном режиме домена).

Создание учетной записи пользователя.

Для создания в домене учетной записи пользователя, предположим с идентификатором popov_as, необходимо выполнить следующее:

1. Укажите подразделение, в котором следует создать учетную запись, и нажмите правую кнопку мыши. В появившемся меню выберите команду **Создать | Пользователь.**

2. В окне диалога **Новый объект - Пользователь** (New Object - User) в поле **Имя входа пользователя** (User logon name) введите уникальный идентификатор, в поле **Имя** (First name) - имя пользователя, в поле **Фамилия** (Last name) - фамилию пользователя, в поле **Полное имя** (Full name) автоматически появятся имя и фамилия пользователя (рисунок 3). После ввода всей необходимой информации нажмите кнопку Далее (Next).

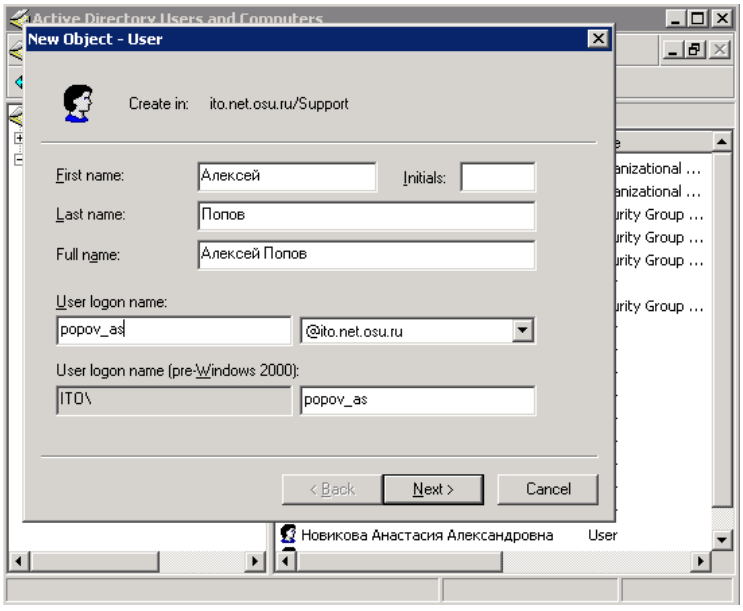

Рисунок 3

3. В следующем окне в полях ввода **Пароль** (Password) и **Подтверждение** (Confirm password) введите с клавиатуры пароль учетной записи пользователя.

4. Если необходима принудительная смена пароля при первой регистрации в сети, установите флажок **Потребовать смену пароля при следующем входе в систему** (User must change password at next logon). С целью защиты от атак по подбору пароля, следует установить срок действия пароля пользователя, сбросив флажок **Срок действия пароля не ограничен** (Password never expires).

6. Установленный флажок **Запретить смену пароля пользователем** (User cannot change password) запрещает пользователю самостоятельно изменять свой пароль.

7. Если только что созданная учетная запись по каким-либо причинам должна быть заблокирована, установите флажок **Отключить учетную запись** (Account disabled).

8. По завершении настройки создаваемой учетной записи нажмите кнопку Далее.

9. В окне диалога, запрашивающего подтверждение правильности выполняемого действия, нажмите кнопку **Готово** (Finish).

Для ввода дополнительной информации или изменения некоторых данных пользователя:

1. Укажите учетную запись пользователя, информацию которой следует изменить, и нажмите правую кнопку мыши. В появившемся меню выберите команду **Свойства.**

2. Внесите необходимые изменения и нажмите кнопку **ОК.**

Перемещение учетной записи пользователя.

Учетную запись пользователя можно перемещать из одного подразделения в другое в пределах одного домена или между доменами. Для соответствующего перемещения учетной записи этого пользователя следует воспользоваться технологией Drag and Drop (перетаскивание), применяемой практически ко всем визуальным объекта операционной системы семейства Windows.

Добавление пользователя в группу.

1. Укажите группу, в которую необходимо добавить пользователя, и нажмите правую кнопку мыши. В появившемся меню выберите команду **Свойства.** Появится окно свойств группы.

2. Перейдите на вкладку **Члены группы** (Members) окна свойств и нажмите **кнопку Добавить.**

3. Появится окно **Выбор: Пользователи, Контакты или Компьютеры** (Select Users, Contacts, or Computers). Здесь можно задать область выполнения запроса: весь каталог, определенный домен или определенная часть дерева подразделения внутри домена. Обратите внимание, что каталог может состоять

из множества доменов.

4. Щелкните на имени добавляемого пользователя и нажмите кнопку **Добавить**. Обратите внимание, что, нажав клавишу <Ctrl> и одновременно выполняя щелчки на нужных объектах, в этом диалоговом окне можно одновременно выбрать несколько пользователей или групп.

В результате все выбранные объекты станут членами соответствующей группы.

Удаленное управление компьютерами.

После создания объекта "компьютер" можно управлять им удаленно, диагностируя службы, работающие на этом компьютере, просматривая события и т. д.

Для того чтобы управлять компьютером удаленно:

1. В окне оснастки **Active Directory- Пользователи и компьютеры** укажите имя компьютера и нажмите правую кнопку мыши. В появившемся меню выберите команду **Управление** (Manage).

2. Для выбранного вами компьютера будет запущена оснастка **Управление компьютером** (Computer Management).

Делегирование прав администрирования.

Как правило, сети больших предприятий на платформе Windows 2000 обладают чрезвычайно разветвленным деревом каталога. Большое количество ветвей, а также наличие достаточно автономных площадок организации, включенных в общее дерево каталога, усложняют управление. Администрирование сети, каталог которой состоит из десятков тысяч объектов, не может безопасно осуществляться одним или несколькими администраторами, имеющими права доступа ко всем объектам.

В подобных случаях следует применять *делегирование прав администрирования.* Это чрезвычайно мощный инструмент, который в больших организациях позволяет более эффективно сконфигурировать систему безопасного администрирования. С его помощью управление отдельными областями сети смогут осуществлять специально назначенные ответственные лица - *администраторы.* При делегировании прав администрирования очень важно наделять ответственных лиц полномочиями, позволяющими выполнять функции администратора только в пределах их зоны ответственности, они не должны иметь возможность администрировать объекты каталога, находящиеся в других частях сети организации.

Права на создание новых пользователей или групп предоставляются на уровне подразделения или контейнера, в котором будут создаваться учетные записи. Администраторы групп одного подразделения могут не иметь прав на создание и управление учетными записями другого подразделения в том же домене. Однако, если права доступа и настройки политик получены на более высоком уровне дерева каталога, они могут распространяться вниз по дереву благодаря механизму *наследования прав доступа.*

Делегирование управления объектами групповой политики.

С помощью инструментов управления Active Directory администратор может делегировать другим пользователям и группам право управления частью каталога. Это в полной мере относится и к объектам групповой политики, в отношении которых могут быть, в частности, делегированы следующие права:

–управление связями GPO с сайтом, доменом или подразделением (организационной единицей). Для этого с помощью инструмента управления Active Directory следует указать объект (сайт, домен или организационную единицу) и щелкнуть правой кнопкой мыши. В появившемся контекстном меню выбрать команду **Делегирование управления** (Delegate Control). Запустится *Мастер делегирования управления* (Delegation of Control Wizard). С его помощью можно выбрать объект групповой политики, группу или пользователя, которому должны быть делегированы права, а также и само право (в данном случае *Управление ссылками групповой политики* (Manage Group Policy links));

–создание и удаление всех дочерних объектов групповой политики. По умолчанию правом создания объектов в GPO обладают администраторы домена (Domain Admins) и администраторы предприятия (Enterprise Admins), а также операционная система. Для делегирования пользователю права управления объектами групповой политики домена необходимо включить его в группу «**Создатели-владельцы групповой политики»** (Group Policy Creator Owners);

–редактирование свойств объектов групповой политики. По умолчанию правом редактирования GPO обладают администраторы домена, администраторы предприятия и операционная система. Для делегирования пользователю права редактирования объекта групповой политики необходимо включить его в одну из указанных групп безопасности.

Чтобы позволить группе или пользователю управлять некоторым подразделением (контейнером):

1. Запустите Active Directory - **Пользователи и компьютеры**.

2. Укажите подразделение, управление которым необходимо передать, и нажмите правую кнопку мыши. В появившемся меню выберите команду **Делегировать управление** (Delegate control). Запустится **Мастер делегирования управления** (Delegation of Control Wizard). Нажмите кнопку **Далее**.

3. В следующем окне мастера нажмите кнопку **Добавить** и выберите пользователя или группу, которой вы хотите разрешить управление подразделением, нажмите кнопку **ОК** и затем кнопку **Далее**.

4. В открывшемся окне диалога мастера делегирования управления в окне со списком **Делегировать следующие обычные задачи** (Delegate the following common tasks) выберите одну или несколько операций, право выполнения которых делегируется указанному пользователю или группе. Если нужно делегировать право выполнения более специализированной задачи, установите переключатель **Создать особую задачу для делегирования** (Create a custom task to delegate). Нажмите кнопку **Далее**.

5. Если указана особая задача для делегирования в следующем окне, можно выбрать область применения для этой задачи: положение переключателя **Этой папкой и существующими в ней объектами, созданием новых объектов в этой папке** (This folder, existing objects in this folder, and creation of new

objects in this folder), в этом случае вы передадите группе право на администрирование всего контейнера, или положение **Только следующими объектами в этой папке** (Only the following objects in the folder) и установить флажки возле нужных объектов, в этом случае группа сможет управлять только *выбранными объектами.* Затем нажмите кнопку **Далее**.

6. В открывшемся окне определяются делегируемые разрешения. Можно отображать и устанавливать *общие* разрешения или разрешения для *отдельных* свойств или *дочерних* объектов. В пределах контейнера можно делегировать не все, а только некоторые права администрирования: например, можно делегировать только права на модификацию (чтение-запись) выбранного контейнера без дочерних объектов. Задайте нужные разрешения и нажмите кнопку **Далее**.

7. В следующем окне сводки выводится информация о выбранных действиях. Можно вернуться назад и скорректировать параметры. Если все правильно, нажмите кнопку **Готово**.

3.3 Вопросы к лабораторной работе № 3

1. Назначение Active Directory и основные возможности.
2. Какова структура Active Directory?
3. Для чего используются организационные единицы, когда и с какой целью их следует создавать?
4. Какие группы пользователей операционная система Windows 2000 создает по умолчанию?
5. Может ли один и тот же пользовать входить в разные группы?
6. В каких случаях следует использовать делегирование прав?
7. Какие права могут быть делегированы?
8. Пользователям каких групп можно делегировать права?

3.4 Задания к лабораторной работе № 3

1. Создать новую организационную единицу (имя выбрать произвольно, например, my_unit).
2. Создать новую группу.
3. Создать в организационной единице трех новых пользователей: для всех потребовать смену пароля при входе и ограничить срок действия пароля. Одного из пользователей включить в новую группу.
4. Делегировать права на созданную организационную единицу пользователю из новой группы.

4 Лабораторная работа № 4. Изучение и настройка политики безопасности операционной системы Windows 2000

4.1 Постановка задачи

1 Изучить структуру и возможности групповых и локальных политик безопасности.

2 Научиться настраивать политику безопасности:

–политику учетных записей;

–глобальную и локальную политики безопасности;

–журнал событий и другие средства защиты и администрирования.

4.2 Теоретические предпосылки

Эффективное функционирование многопользовательской операционной системы невозможно без четкого разграничения доступа к ресурсам. Одним из средств, позволяющих настраивать параметры безопасной работы пользователей в сети в операционных системах семейства Windows (NT, 2000, XP и выше), являются *политики безопасности.*

Реализация политик безопасности в Windows 2000 предоставляет достаточно широкие возможности, в том числе настройку политики безопасности для всего дерева доменов. Установив политику безопасности в одном месте, администраторы могут контролировать безопасность всех рабочих станций домена. Политики безопасности в Windows 2000 реализуются с помощью средств *групповых политик* (group policy).

Групповая политика имеет следующие преимущества:

–основываясь на службе Active Directory системы Windows 2000, позволяет как централизованно, так и децентрализовано управлять параметрами политики;

–обладает гибкостью и масштабируемостью. Может быть применена в широком наборе конфигураций системы, предназначенных как для малого бизнеса, так и для больших корпораций;

–обладает высокой степенью надежности и безопасности;

–групповые политики расширяют и используют преимущества Active Directory. Их настройки находятся в *объектах групповых политик* (Group Policy Object, GPO), которые в свою очередь ассоциируются с такими контейнерами Active Directory, как сайты, домены и подразделения (организационные единицы).

Для запуска объекта **Групповая политика** (рисунок 4) следует выполнить следующие действия:

1) выбрать объект Active Directory, для которого необходимо установить групповую политику безопасности (на рисунке 4 – для всего контролера доме-

на) и правой кнопкой мыши вызвать контекстное меню;

2) выбрать элемент **Свойства** (Properties)**;**

3) в диалоговом окне свойств выбрать вкладку **Групповая политика** (Group Policy);

4) для модификации глобальной политики следует выбрать кнопку **Edit**, если политика еще не была создана – **New** и ввести имя, либо воспользоваться тем, которое предлагает система.

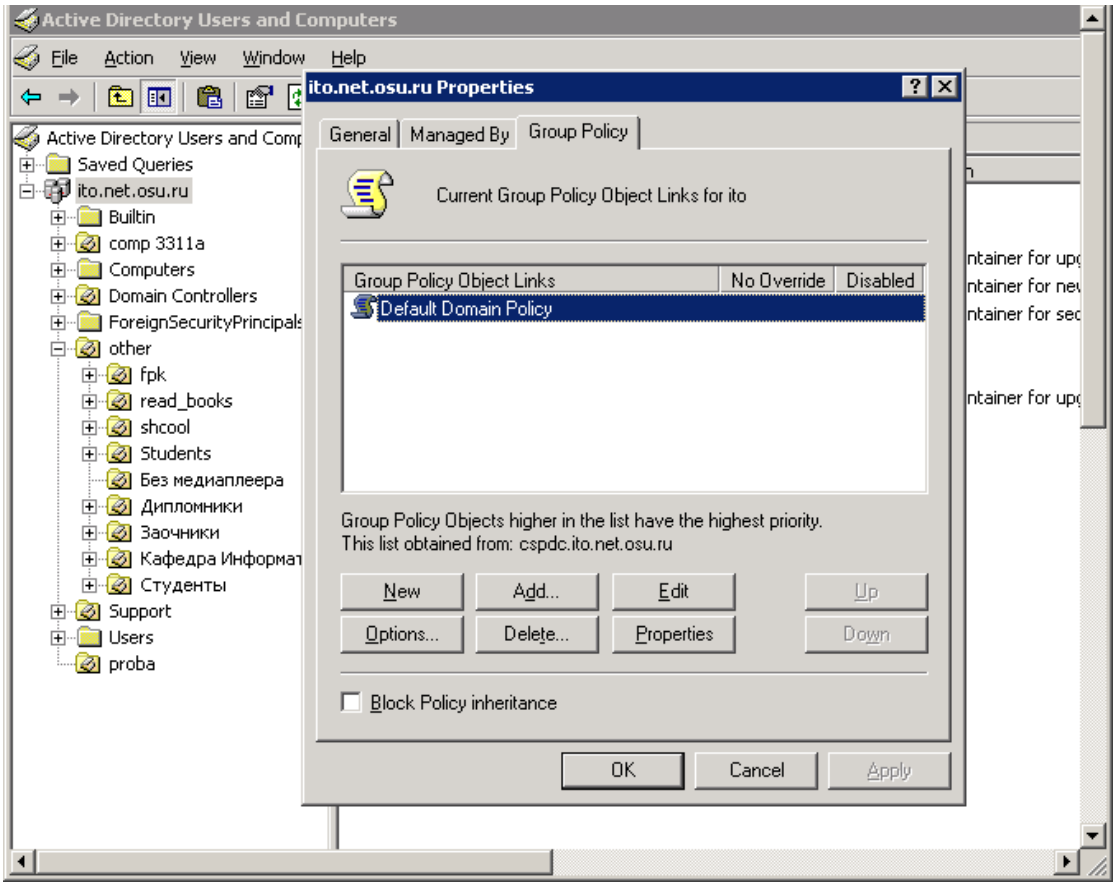

Рисунок4

Создать групповую политику для контейнера Active Directory можно только при наличии определенного набора условий. Необходимо иметь работающий контроллер домена Windows 2000. Пользователь, который создает групповую политику, должен обладать правами на чтение и запись в системный том контроллеров домена (папка Sysvol). Кроме того, он должен иметь право модификации выбранного контейнера Active Directory.

После выбранных действий загружается корневой узел (рисунок 5), представляющий собой GPO, присоединенный к определенному контейнеру.

Имя этого GPO и имя контейнера, к которому он присоединен, отображаются в окне структуры в следующем формате:

Имя_политики [Имя_домена] Policy

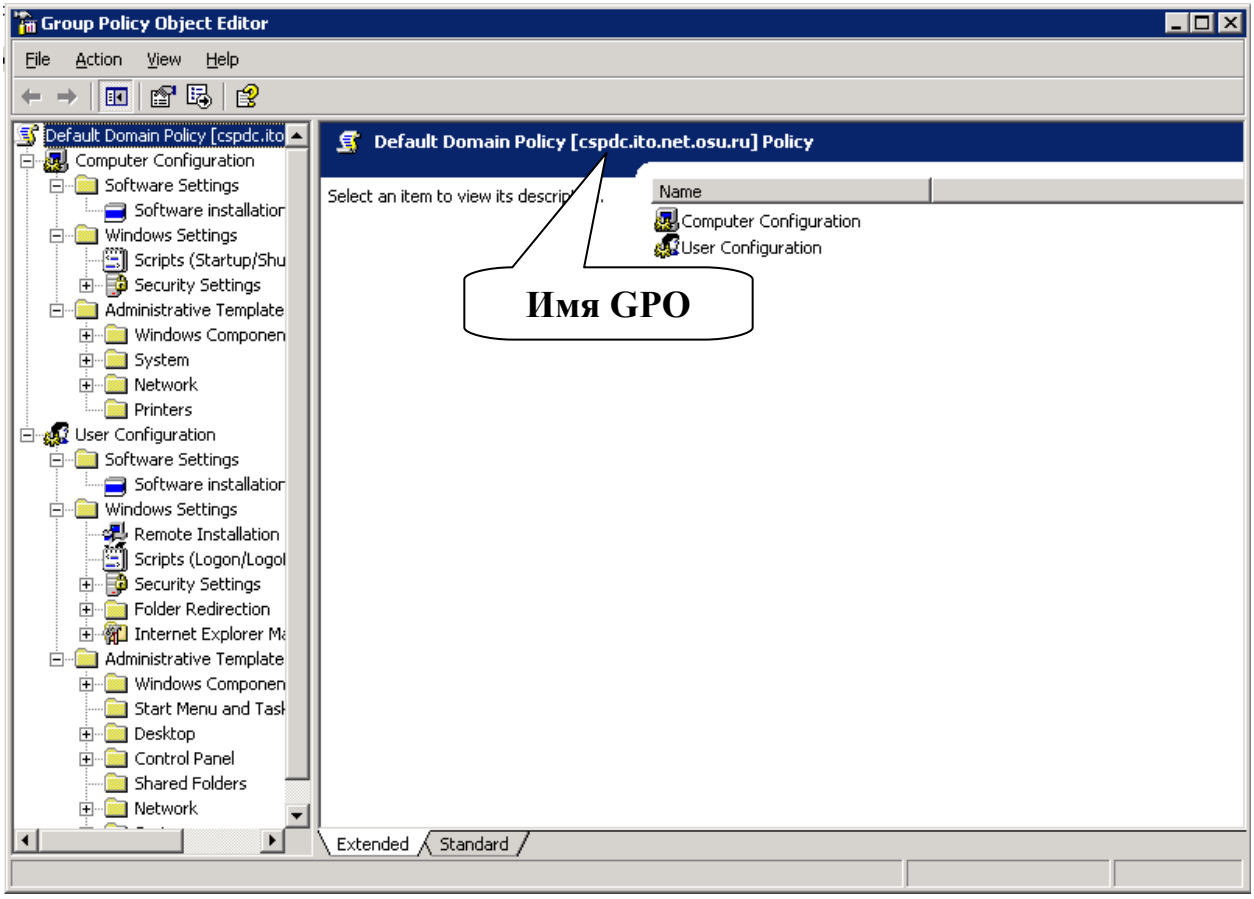

Рисунок 5

Затем пространство имен подразделяется на два узла более низкого уровня: «**Конфигурация компьютера»** (Computer Configuration) и «**Конфигурация пользователя»** (User Configuration). Используя их, можно создавать и настраивать групповые политики для компьютера и пользователей.

Узел «**Конфигурация компьютера»** содержит параметры всех политик, определяющих работу *компьютера.* Они регулируют функционирование операционной системы, вид рабочего стола, задают параметры выполняемых приложений, определяют работу средств обеспечения безопасности и т. д. Групповая политика применяется к рабочей станции домена на этапе загрузки системы и в дальнейшем при выполнении циклов обновления.

Узел **«Конфигурация пользователя»** содержит параметры всех политик, определяющих работу *пользователя* на компьютере. Они регулируют вид рабочего стола как и в предыдущем случае, задают параметры выполняющихся приложений, определяют работу средств обеспечения безопасности и пользовательских сценариев входа и выхода. Групповая политика применяется к пользователю при его регистрации и в дальнейшем при выполнении циклов обновления.

Опишем некоторые расширения объекта «**Групповая политика»**:

–административные шаблоны. (Administrative Templates). Здесь находится групповая политика, определяющая параметры реестра, задающие работу и внешний вид рабочего стола, компонент операционной системы и приложе-

ний;

–*параметры безопасности* (Security Settings). Служит для настройки параметров системы безопасности компьютеров, на которые воздействует данный объект групповой политики. С помощью групповых политик можно настроить безопасность индивидуального компьютера, домена и целой сети;

–*установка программ* (Software Installation). Служит для централизованного управления программным обеспечением организации. С его помощью можно задавать различные режимы установки новых программ на компьютеры пользователей;

–*сценарии* (Scripts). Сценарии используются для автоматического выполнения набора команд при загрузке операционной системы и в процессе завершения ее работы, а также при регистрации и отключении пользователя от сети. Для выполнения сценариев, написанных на Microsoft JScript и Microsoft Visual Basic Scripting Edition, можно применять сервер сценариев (Windows Scripting Host);

–*перенаправление папок* (Folder Redirection). Позволяет перенаправлять обращение к специальным папкам в сеть.

С помощью расширения «**Параметры безопасности»** (рисунок 6) в GPO можно определить параметры политики безопасности, определяющие различные аспекты работы системы безопасности Windows 2000. Созданная в объекте групповой политики конфигурация воздействует на все компьютеры, находящиеся в контейнере, к которому присоединен данный GPO.

Расширение «**Параметры безопасности»** позволяет настраивать следующие аспекты системы безопасности компьютера:

–*политики учетных записей* (Account Policies). Можно настраивать политики безопасности как учетных записей в масштабах домена, так и локальных учетных записей. Здесь определяются политика паролей, политика блокировки паролей и политика Kerberos, распространяющаяся на весь домен;

–*локальные политики* (Local Policies). Можно настраивать политику аудита, назначать права пользователей и различные параметры безопасности, доступные для настройки в системе Windows 2000;

–*журнал событий* (Event Log). Можно настраивать политики безопасности, определяющие работу журналов событий приложений, системы и безопасности;

–*группы с ограниченным доступом* (Restricted Groups). Можно регулировать членство пользователей в специфических группах. Сюда обычно включают встроенные группы, такие как Администраторы, Операторы архива и другие, имеющие по умолчанию права администратора. В эту категорию могут быть включены и другие группы, безопасность которых требует особого внимания и членство в которых должно регулироваться на уровне политики;

–*системные службы* (System Services). Можно настраивать безопасность и параметры загрузки для работающих на компьютере служб. В этом разделе могут быть использованы расширения, с помощью которых можно осуществлять настройку безопасности, специфическую для данной службы. Напри-

мер, расширение File Sharing Service позволяет настраивать политику безопасности для службы создания общего доступа к файлу (ограничение анонимного доступа к общим ресурсам, формирование безопасности различных сетевых общих ресурсов и т. д.);

–*реестр* (Registry). Можно настраивать безопасность различных разделов реестра;

–*файловая система* (File System). Можно настраивать безопасность определенных файлов;

–*политики открытого ключа* (Public Key Policies). Можно настраивать политики безопасности в отношении шифрования информации с помощью EFS, авторизации корневого сертификата в масштабах домена, авторизации доверенного сертификата и т. д.;

–*политики безопасности IP* (IPSEC). Позволяет настраивать политику безопасности IP для компьютеров.

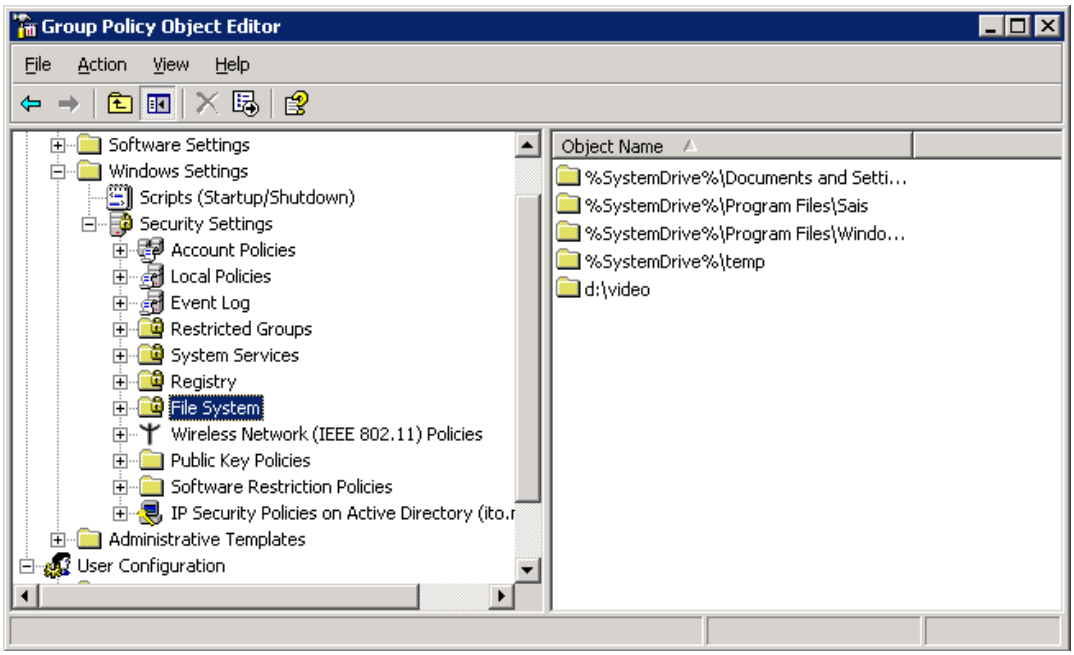

Рисунок 6

Политики безопасности, определяемые расширением «**Параметры безопасности»**, действуют на компьютеры и частично на пользователей. Поскольку политика безопасности Windows 2000 значительно отличается от политик предыдущих версий Windows NT, при переходе к Windows 2000 низкоуровневые политики безопасности не переносятся. Если при переходе создается новое дерево доменов, одновременно создается и новая политика безопасности, назначаемая по умолчанию. Если при переходе домен присоединяется к уже существующему дереву, политика безопасности берется от родительского домена.

Для модификации настроек безопасности щелкните на папке «**Параметры безопасности»,** затем щелчками на соответствующих узлах откройте весь путь, ведущий к интересующим настройкам. В правом подокне окна «**Групповая политика»** двойным щелчком выберите настраиваемую политику и в

открывшемся окне настройте ее.

Рассмотрим работу указанных расширений на конкретных примерах.

1. Настройка политики паролей.

Предположим, нам необходимо установить следующие правила политики паролей и блокировки:

–минимальная длина пароля – 8 символов;

–максимальный срок действия пароля – 30 дней;

–блокировать консоль после трех неудачных попыток входа.

Для реализации указанных правил выполним следующие действия:

1) откроем глобальную политику безопасности домена (см. выше) и расширение **«Политики безопасности»** (Security Settings);

2) выберем пункт **«Политика учетных записей»** (Account Policies), а затем политику паролей (Password Policy);

3) в правой части окна появится полный список правил, поддерживаемых политикой безопасности Windows 2000;

4) найдем требуемые правила:

а) минимальная длина пароля (Minimum password length);

б) максимальный срок действия пароля (Maximum password age);

установим требуемые значения, вызвав соответствующие диалоговые окна двойным щелчком мыши на названии правила;

5) для установки параметра блокировки перейдем в раздел политики блокировки учетных записей (Account Lockout Policy), выберем необходимое правило в правой части окна – Account lockout threshold – и установим требуемое значение – три.

2. Политика учетных записей.

Предположим, нам необходимо разрешить всем пользователям домена использовать привилегию изменения системного времени. Для этого следует выполнить:

1) откроем глобальную политику безопасности домена (см. выше) и расширение **«Политики безопасности»** (Security Settings);

2) выберем пункт «Локальные политики» (Local Policies), а затем политику назначения прав пользователей (User Rights Assignment);

3) в правой части окна выберем требуемое правило – Change the system time (рисунок 7) и добавим пользователя Все (Evryone).

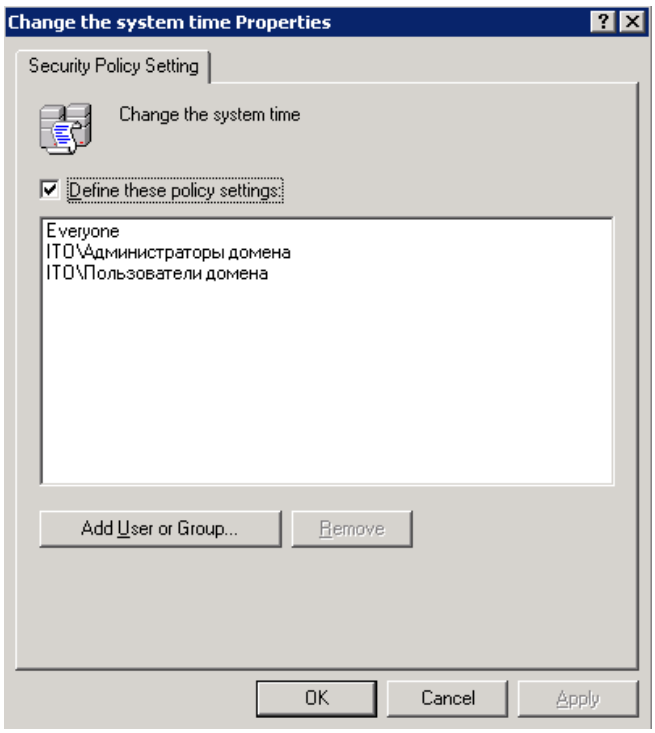

Рисунок 7

Применение групповых политик происходит в последовательности, соответствующей иерархии GPO: сначала объект групповой политики сайта, затем домена, затем GPO, связанные с подразделениями в соответствии с их вложенностью. Порядок выполнения групповых политик можно изменить с помощью настроек, блокирующих определенные групповые политики или заставляющих их выполняться принудительно. Кроме того, на порядок выполнения групповых политик влияет применение групп безопасности.

По умолчанию настройки групповой политики, применяемые к контейнеру определенного уровня, наследуются всеми контейнерами более низких уровней и находящимися внутри них пользователями и компьютерами. Если с дочерней организационной единицей (контейнером) связан свой GPO, он может устанавливать для нее *индивидуальные* настройки групповых политик, отменяющие применение к ней *наследуемых* настроек. Если некоторые настройки групповых политик родительского контейнера не заданы (not defined), то они не наследуются и дочерними контейнерами. Если родительский контейнер обладает сконфигурированными настройками групповых политик, которые *не* заданы в GPO дочернего контейнера, то такие настройки наследуются.

Наследование настроек групповых политик родительского контейнера дочерним контейнером, с которым связан собственный объект групповой политики, может иметь место только в случае *совместимости* этих групповых политик. Например, если политика родительского контейнера задает определенную конфигурацию рабочего стола компьютера пользователя, а политика дочернего контейнера дополняет ее, пользователь увидит на своем рабочем столе все элементы, заданные обеими политиками. Если же групповая политика родительского контейнера противоречит групповой политике дочернего контейнера, выполняются только настройки GPO, связанного с дочерним контейнером.

Подобное положение вещей может быть изменено. Установка флажка **Блокировать наследование политики** (Block Policy inheritance), находящегося на вкладке **Групповая политика** окна свойств некоторого контейнера, *запрещает наследование* каких-либо групповых политик, установленных для *родительского* контейнера.

Существует средство, позволяющее настроить *принудительное применение* групповой политики, настроенной для некоторого контейнера, всеми контейнерами *более низкого* уровня. Для этого на вкладке **Групповая политика** окна свойств контейнера следует нажать кнопку **Параметры** (Options). В появившемся окне диалога **Параметры *<имя_подразделения>*** необходимо установить флажок **Не перекрывать** (No override). В этом случае дочерние контейнеры будут наследовать (т. е. не смогут переопределить) все настройки родительского контейнера, даже в том случае, если для дочерних контейнеров установлен флажок **Блокировать наследование политики.**

По умолчанию групповая политика применяется синхронно, т. е. политики компьютера применяются до появления окна «**Вход в Windows»** (Log on to Windows), а политики пользователя – до передачи операционной системой управления оболочке, интерактивно взаимодействующей с пользователем. Подобный порядок можно изменить, однако делать это не рекомендуется, поскольку асинхронное применение групповых политик может привести к непредсказуемым и нежелательным результатам.

Применение групповых политик не ограничивается только, например, моментом загрузки операционной системы компьютера или регистрацией пользователя в системе. При работе компьютера в сети групповые политики могут измениться, поэтому они применяются периодически (по умолчанию – каждые 90 минут). Длительность периода применения политик можно изменять. Если задать его равным нулю, групповые политики применяются через каждые 7 секунд. Следует учитывать, что при уменьшении периода применения групповых политик значительно увеличивается нагрузка на систему. На контроллерах доменов период применения политик равен 5 минутам.

Настройки расширений **Установка программ и Переназначение папки** применяются только при загрузке операционной системы или регистрации пользователя в системе, поскольку периодическое применение этих групповых политик может вызвать нежелательные результаты.

4.3 Вопросы к лабораторной работе № 4

Какие виды политик безопасности поддерживаются в Windows 2000, сферы их применения?

1. Какие параметры безопасности можно настроить в глобальной политике безопасности?
2. Как взаимодействуют между собой глобальная и локальная политики безопасности?
3. Какие правила наследования политик безопасности поддерживаются?

4.4 Задания к лабораторной работе № 4

1. Установить максимальный срок действия пароля – 30 дней.
2. При вводе нового пароля требовать его неповторяемость. Хранить в системе 2 предыдущих пароля.
3. Установить минимальную длину пароля – 10 символов.
4. Установить аудит успеха для событий входа в систему.
5. Назначить возможность выключения системы только для администраторов.
6. Разрешить вход в систему только для членов группы "Администраторы" и пользователя "_____".
7. Разрешить доступ к компьютеру из сети только для пользователя "_____".
8. Блокировать консоль пользователя после ввода двух неверных паролей на 5 минут.
9. Отображать последнее имя пользователя при диалоге входа в систему.
10. Разрешить пользователю "_____" изменять политику аудита системы.

Список использованных источников

1 **Виноградов И.М.** Элементы высшей математики: учеб. для вузов / И.М. Виноградов -М.: Высш. шк., 1999.

2 **Зегжда Д.П.** Основы безопасности информационных систем: учеб. пособие / Д.П. Зегжда, А.М. Ивашко . - М. : Горячая линия - Телеком, 2000.

3 **Домарев В.В. Защита информации и безопастность компьютерных систем** / В.В. Домарев. -Киев : Диа-Софт, 1999. - 480 с.

4 **Романец Ю.В.** Защита информации в компьютерных системах и сетях / Ю.В. Романец, П.А. Тимофеев, В.Ф. Шаньгин; под ред. В.Ф. Шаньгина.- 2-е изд., перераб. и доп. -М. : Радио и связь, 2001.

5 **Жельников В.** Криптография от папируса до компьютера / В. Жельников. -М. : ABF, 1996. - 336с.

6 **Проскурин В.Г.** Программно-аппаратные средства обеспечения информационной безопасности. Защита в операционных системах: учеб. пособие для вузов / Проскурин В.Г., Крутов С.В., Мацакевич И.В. - М.: Радио и связь, 2000.

7 **Зубанов Ф.В.** Microsoft Windows 2000. Планирование, развертывание, установка / Ф.В. Зубанов. – 2-ое изд., испр. – М.: Издательско-торговый дом «Русская редакция», 2000. - 592 с.