



8 Олимпиада школьников по информатике и компьютерной безопасности

ВАРИАНТ 1

Задача 1. Подбор пароля

В результате анализа подсистемы аутентификации сервера террористов удалось выяснить, что подтверждение имени пользователя выполняется с помощью паролей. Было установлено, что при проверке введенный пароль подвергается преобразованию при помощи функции, восстановленный исходный код которой приведен ниже (см. табл.). Для принятия решения о том верен ли пароль, результат работы функции сверяется с эталонами, хранимыми на сервере. Найдите пароль, который пройдет проверку, если известно, что пароль «БЕЗОПАСНОСТЬ» верен.

C	Pascal
<pre>int HASH(char *text) { int k=0, H; char temp[4]; for (int i=0; (i<strlen(text))&&(k<4); i++) if ((i+1)%2==0) {temp[k]=text[i]; k++;} if (k<4) for(;k<4;k++) temp[k]='D'; H = ((temp[0]+temp[3])&255)*256; H += (temp[1]+temp[2])&255; return H; }</pre>	<pre>function HASH (text: string):Integer; var i, k, HA : Integer; temp: array [1..4] of char; begin k := 1; i := 1; while ((i <= Length (text)) and (k <= 4)) do begin if (i mod 2 = 0) then begin temp[k] := text[i]; k := k+1; end; i := i+1; end; if (k <= 4) then begin while k <= 4 do begin temp[k] := 'D'; k := k+1; end; end; HA := ((ord(temp[1])+ord(temp[4])) and 255)*256; HA := HA + ((ord(temp[2])+ord(temp[3])) and 255); HASH := HA; end;</pre>

Фрагмент таблицы ASCII

hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	“	#	\$	%	&	‘	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	:	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL
8	A	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П
9	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я

Задача 2. Доступ

Специалисты по информационной безопасности, проанализировав компьютерную систему, пришли к следующим выводам:

1. В системе хранятся файлы F_1, F_2, F_3 ;
2. Система имеет в своем составе набор программ S_0, S_1, \dots, S_6 , которые кооперируют друг с другом при передаче прав доступа.
3. Все элементы компьютерной системы обладают некоторым (возможно пустым) набором прав доступа, что представлено в виде матрицы доступов.

Выясните, возможно ли чтение данных программой S_0 из файла F_1 ? Ответ обосновать.

	F_1	F_2	F_3	S_0	S_1	S_2	S_3	S_4	S_5	S_6
S_0										
S_1						g			t	
S_2							t			
S_3										t
S_4				t						
S_5								t		
S_6	r									

Комментарий

Каждая строка матрицы доступов описывает права одной программы в системе. Каждый элемент строки описывает, какими правами обладает программа по отношению к элементу системы, которым помечен соответствующий столбец. Например, в приведенной таблице программа S_6 имеет право на чтение файла F_1 . В общем случае, программа может иметь несколько прав к одному и тому же элементу системы.

Право r (read) показывает, что программа может обратиться и считать данные, относящиеся к элементу. Права, обозначенные как t (take) и g (grant), являются соответственно правом брать право и давать право. Обладая этими правами, программы могут изменять набор прав доступа согласно правилам, приведенным в таблице.

Название правила	Состояние элементов таблицы доступов до применения правила	Состояние элементов таблицы доступов после применение правила																								
Take	<table border="1"> <thead> <tr> <th></th> <th>O</th> <th>S_1</th> <th>S_2</th> </tr> </thead> <tbody> <tr> <th>S_1</th> <td>β</td> <td></td> <td></td> </tr> <tr> <th>S_2</th> <td></td> <td>t</td> <td></td> </tr> </tbody> </table> <p>S_i – программа; O – файл или программа; β – любое право доступа</p>		O	S_1	S_2	S_1	β			S_2		t		<table border="1"> <thead> <tr> <th></th> <th>O</th> <th>S_1</th> <th>S_2</th> </tr> </thead> <tbody> <tr> <th>S_1</th> <td>β</td> <td></td> <td></td> </tr> <tr> <th>S_2</th> <td>β</td> <td>t</td> <td></td> </tr> </tbody> </table>		O	S_1	S_2	S_1	β			S_2	β	t	
	O	S_1	S_2																							
S_1	β																									
S_2		t																								
	O	S_1	S_2																							
S_1	β																									
S_2	β	t																								
Grant	<table border="1"> <thead> <tr> <th></th> <th>O</th> <th>S_1</th> <th>S_2</th> </tr> </thead> <tbody> <tr> <th>S_1</th> <td></td> <td></td> <td></td> </tr> <tr> <th>S_2</th> <td>β</td> <td>g</td> <td></td> </tr> </tbody> </table> <p>S_i – программа; O – файл или программа; β – любое право доступа</p>		O	S_1	S_2	S_1				S_2	β	g		<table border="1"> <thead> <tr> <th></th> <th>O</th> <th>S_1</th> <th>S_2</th> </tr> </thead> <tbody> <tr> <th>S_1</th> <td>β</td> <td></td> <td></td> </tr> <tr> <th>S_2</th> <td>β</td> <td>g</td> <td></td> </tr> </tbody> </table>		O	S_1	S_2	S_1	β			S_2	β	g	
	O	S_1	S_2																							
S_1																										
S_2	β	g																								
	O	S_1	S_2																							
S_1	β																									
S_2	β	g																								
Create	<table border="1"> <thead> <tr> <th></th> <th>S</th> </tr> </thead> <tbody> <tr> <th>S</th> <td></td> </tr> </tbody> </table> <p>S – программа;</p>		S	S		<table border="1"> <thead> <tr> <th></th> <th>O</th> <th>S</th> </tr> </thead> <tbody> <tr> <th>S</th> <td>β</td> <td></td> </tr> </tbody> </table> <p>S – программа; O – файл или программа;</p>		O	S	S	β															
	S																									
S																										
	O	S																								
S	β																									

Задача 3. Сеть

В одноэтажном здании стоит 2013 компьютеров. Они объединены в последовательную сеть топологии кольцо. План расположения кабелей – это 2013-угольный невыпуклый многоугольник. Нарушители имеют в распоряжении этот план и у них имеется возможность, не вызывая подозрений, проложить кабель по одной прямой линии для подсоединения к проводам сети (прямая не проходит через точки подсоединения компьютеров).

Смогут ли нарушители проложить кабель так, чтобы полностью контролировать обмен информации в сети? Ответ обоснуйте.

Задача 4. Путаница

Вася совсем запутался и не может понять, что делает функция g, которую написал Петя. Объясните ему, каким образом и что она делает (единственное, что Вася точно знает так это то, что на вход функции g Петя подает положительные значения m1 и m2).

```
struct k{int m[4];};k m(k m1, k m2){return k{
    m1.m[0]*m2.m[0]+m2.m[2]*m1.m[1],
    m2.m[1]*m1.m[0]+m1.m[1]*
    m2.m[3], m2.m[0]*
    m1.m[2]+m2.m[2] *m1.m[3],
    m2.m[1]*m1.m[2]+m1.m[3]*m2.m[3]};}
int g(int n) {k p={0,1,1,1};k r={1,0,0,1};while(n)
{ if(n&1)r=m(r, p); p=m(p,p);n >>=1;}return r.m[ 2 ] ; }
```

Задача 5. Фильтр

Специалистам по информационной безопасности удалось подключиться к локальной сети террористической организации. Получив возможность анализа трафика в сети, было решено выяснить имена всех учётных записей, которыми пользуются террористы в одной из социальных сетей. Для перехвата данных использовалась специальная программа (сниффер). Объем обрабатываемого трафика очень велик, и обработать его вручную не представляется возможным, поэтому необходимо воспользоваться возможностями сниффера по анализу содержимого html-страниц с применением регулярных выражений. Напишите регулярное выражение, позволяющее выделить искомый логин.

Код типовой страницы авторизации имеет следующий вид:

```
<html>
<head>
  <title>Самая лучшая социальная сеть</title>
</head>
<body>
  <form>
    <input type="hidden" name="login"
      value="vasyaisthebestofthebest@mail.ru" />
    <input type="text" id="user56789" name="login"
      value="vasyaisthebestofthebest@mail.ru" class="loginstring"/>
    <input type="hidden" name="password" value="bestman12"/>
    <input type="password" name="password" class="passwordstring"/>
    <input type="submit" name="action" value="Войти на сайт!" />
```

</form>
</body>

Комментарий

Регулярные выражения предоставляют возможности для описания подстрок определенного вида. Для формирования регулярного выражения используются следующие элементы:

1. символ – например, «a»
2. любой символ – обозначается «.»
3. пробельный символ – обозначается «\s»
4. диапазон символов – обозначается «[]». Например:
[abc] – любая из букв a, b, c
[a-z0-9] – любая из малых букв латинского алфавита и цифра
5. отрицание диапазона:
[^a-z5] – не маленькая буква латинского алфавита и цифра 5

Для указания количества вхождений используются кванторы:

- «?» означает «0 или 1 шт.»
- «+» означает «>= 1 шт.» , причем берется как можно большее количество символов
- «+?» означает «>= 1 шт.» , причем берется как можно меньшее количество символов
- «*» означает «>= 0 шт.» , причем берется как можно большее количество символов
- «*?» означает «>= 0 шт.» , причем берется как можно меньшее количество символов
- «{5}» означает «5 шт.»
- «{5, 8}» означает «от 5 до 8 шт.»

Например, «a*» – любое количество идущих подряд символов «a».

Для извлечения подстроки используются круглые скобки, например, выражение: $([0-9]^+)\s+([0-9]^+)$ сохраняет из строки «мои числа 45 567 234 56» числа: «45» и «567».