

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**

**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО**

**ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ**

**«Донской государственный технический университет»**

**(ДГТУ)**

Кафедра «Программное обеспечение вычислительной техники и

автоматизированных систем»

**ЛАБОРАТОРНАЯ РАБОТА № 3**

по дисциплине «Защита информации»

тема: «Дискреционная модель политики безопасности»

Выполнил:

ст. гр. ВПР 31 Д. С. Кононов

Проверил:

д.н., профессор Е. А. Романенко

Ростов-на-Дону

2020

**Вариант №13**

Цель лабораторной работы: реализовать программный модуль, демонстрирующий работу пользователя в дискреционной модели политики безопасности.

Количество субъектов доступа = 7

Количество объектов доступа = 4

**Текст программы**

from random import randint

class Chmod:

def \_\_init\_\_(self, entity=4):

self.rights\_name = ['Grant', 'Write', 'Read']

self.rights = [[0, 0, 0] for i in range(entity)]

self.entity = [randint(0, 100) for i in range(entity)]

def write(self, num, new):

if self.rights[num][1] == 1:

self.read(num)

old = self.entity[num]

self.entity[num] = new

print('Entity {} = {} has been changed to {}'.format(num, old, self.entity[num]))

else:

print('Dont have rights to write')

def read(self, num):

if self.rights[num][2] == 1:

print('Entity {} ='.format(num), self.entity[num])

else:

print('Dont have rights to read')

def change\_right(self, num, right, new\_right):

if self.rights[num][0] == 1:

self.rights[num][right] = new\_right

print('Right successful changed')

else:

print('Dont have rights to grant')

class User(Chmod):

def \_\_init\_\_(self, admin=1, entity=4):

super().\_\_init\_\_()

print('Initialization was successful')

if admin == 0:

self.rights = [[1, 1, 1] for i in range(entity)]

else:

self.rights = [[randint(0, 1), randint(0, 1), randint(0, 1)] for j in range(entity)]

for j in self.rights:

if j[-1] == 0 and j[-2] == 1:

j[-1] = 1

def \_\_str\_\_(self):

string = 'Your roots:\n'

for i in range(len(self.rights)):

string += 'Entity {}: '.format(i)

c = 0

tmp = ''

for j in range(len(self.rights\_name)):

if self.rights[i][j]:

c += 1

tmp += self.rights\_name[j] + ' '

if c == 0:

string += 'No rights'

elif c == 3:

string += 'All rights'

else:

string += tmp

string += '\n'

return string

if \_\_name\_\_ == '\_\_main\_\_':

while True:

user = User(admin=int(input('Enter a user: ')))

print(user)

while True:

print('1) Show my roots',

'2) Read entity',

'3) Write to entity',

'4) Change right',

'0) Quit',

sep='\n')

choice = input('Enter an action: ')

if choice == '1':

print(user)

elif choice == '2':

user.read(int(input('Enter index entity: ')))

elif choice == '3':

user.write(int(input('Enter index entity: ')), int(input('Enter new value to entity: ')))

elif choice == '4':

user.change\_right(

int(input('Enter index entity: ')),

int(input('Enter index right: ')),

int(input('Enter new right for entity: '))

)

elif choice == '0':

break