Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по курсу **«Защита информации в сети Internet»**

по лабораторной работе №5

на тему «**Атрибуты безопасности в ОС Windows**»

Выполнили  
студенты группы 18ВВ1:  
 Кривозубов А. К.

Махров А. А.

Приняли:  
к.т.н., доцент Дубравин А.В.

Пенза 2020

**Цель работы:**

Изучить методы работы с атрибутами безопасности в ОС Windows.

**Задание:**

|  |  |
| --- | --- |
| Вариант | Задание |
| 6 | Разработать программу, выполняющую создание раздела реестра «HKEY\_CURRENT\_USER\Software\pi» с атрибутами безопасности для текущего пользователя и правами только чтение, а для группы «Администраторы» на полный доступ. |

**Листинг программы:**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <Windows.h>

#include <AccCtrl.h>

#include <Aclapi.h>

#include <stdio.h>

#include <conio.h>

void error(const char \*error) {

printf("%s\n", error);

\_getch();

exit(1);

}

int main(int argc, char\* argv[])

{

// Объявляем структуру EXPLICIT\_ACCESS для двух групп

EXPLICIT\_ACCESS ea[2];

// Предварительно обнуляем

ZeroMemory(&ea, 2 \* sizeof(EXPLICIT\_ACCESS));

//Получение идентификатора группы Everyone

SID\_IDENTIFIER\_AUTHORITY SIDAuthWorld = SECURITY\_WORLD\_SID\_AUTHORITY;

PSID pEveryoneSID = NULL;

if (!AllocateAndInitializeSid(&SIDAuthWorld, 1, SECURITY\_WORLD\_RID, 0, 0, 0, 0, 0, 0, 0, &pEveryoneSID)) {

error("AllocateAndInitializeSid Error");

}

// Заполнение структуры EXPLICIT\_ACCESS для группы Everyone

ea[0].grfAccessPermissions = KEY\_READ;

ea[0].grfAccessMode = SET\_ACCESS;

ea[0].grfInheritance = NO\_INHERITANCE;

ea[0].Trustee.TrusteeForm = TRUSTEE\_IS\_SID;

ea[0].Trustee.TrusteeType = TRUSTEE\_IS\_WELL\_KNOWN\_GROUP;

ea[0].Trustee.ptstrName = (LPTSTR)pEveryoneSID;

//Получение идентификатора группы Администратор

SID\_IDENTIFIER\_AUTHORITY SIDAuthNT = SECURITY\_NT\_AUTHORITY;

PSID pAdminSID = NULL;

if (!AllocateAndInitializeSid(&SIDAuthNT, 2, SECURITY\_BUILTIN\_DOMAIN\_RID, DOMAIN\_ALIAS\_RID\_ADMINS, 0, 0, 0, 0, 0, 0, &pAdminSID)) {

FreeSid(pEveryoneSID);

error("AllocateAndInitializeSid Error");

}

// Заполнение структуры EXPLICIT\_ACCESS для группы Администратор

ea[1].grfAccessPermissions = KEY\_ALL\_ACCESS;

ea[1].grfAccessMode = SET\_ACCESS;

ea[1].grfInheritance = NO\_INHERITANCE;

ea[1].Trustee.TrusteeForm = TRUSTEE\_IS\_SID;

ea[1].Trustee.TrusteeType = TRUSTEE\_IS\_GROUP;

ea[1].Trustee.ptstrName = (LPTSTR)pAdminSID;

// Генерируется новый список ACL

PACL pACL = NULL;

DWORD dwRes = SetEntriesInAcl(2, ea, NULL, &pACL);

if (dwRes != ERROR\_SUCCESS) {

FreeSid(pEveryoneSID);

FreeSid(pAdminSID);

error("SetEntriesInAcl Error");

}

// Выделяем память под дескриптор безопасности

PSECURITY\_DESCRIPTOR pSD = (PSECURITY\_DESCRIPTOR)LocalAlloc(LPTR, SECURITY\_DESCRIPTOR\_MIN\_LENGTH);

if (pSD == NULL) {

FreeSid(pEveryoneSID);

FreeSid(pAdminSID);

LocalFree(pACL);

error("LocalAlloc Error");

}

// Инициализируем дескриптор безопасности

if (!InitializeSecurityDescriptor(pSD, SECURITY\_DESCRIPTOR\_REVISION)) {

FreeSid(pEveryoneSID);

FreeSid(pAdminSID);

LocalFree(pACL);

LocalFree(pSD);

error("InitializeSecurityDescriptor Error");

}

if (!SetSecurityDescriptorDacl(pSD, TRUE, pACL, FALSE)) {

FreeSid(pEveryoneSID);

FreeSid(pAdminSID);

LocalFree(pACL);

LocalFree(pSD);

error("SetSecurityDescriptorDacl Error");

}

// Теперь, когда дескриптор безопасности создан,

// его можно использовать при создание файла, заполнив структуру SECURITY\_ATTRIBUTES

SECURITY\_ATTRIBUTES sa;

sa.nLength = sizeof(SECURITY\_ATTRIBUTES);

sa.lpSecurityDescriptor = pSD;

sa.bInheritHandle = FALSE;

HKEY hkSub;

DWORD dwDisposition;

// Создание реестра

LONG lRes = RegCreateKeyEx(HKEY\_CURRENT\_USER, ("Software\\pi"), 0, NULL, 0, KEY\_READ | KEY\_WRITE, &sa, &hkSub, NULL);

printf("RegCreateKeyEx result: ");

switch (lRes) {

case ERROR\_SUCCESS:

printf("OK!\n");

break;

default:

printf("ERROR!\n");

}

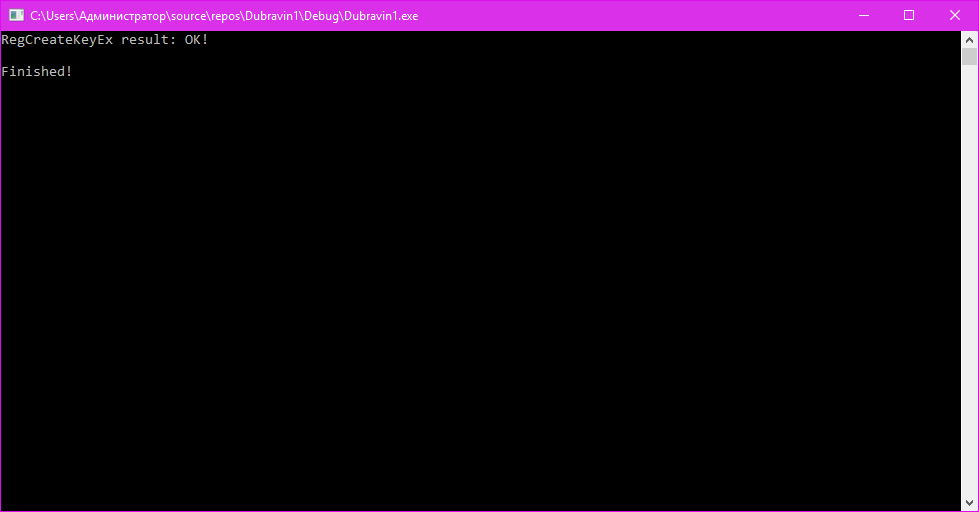
printf("\nFinished!");

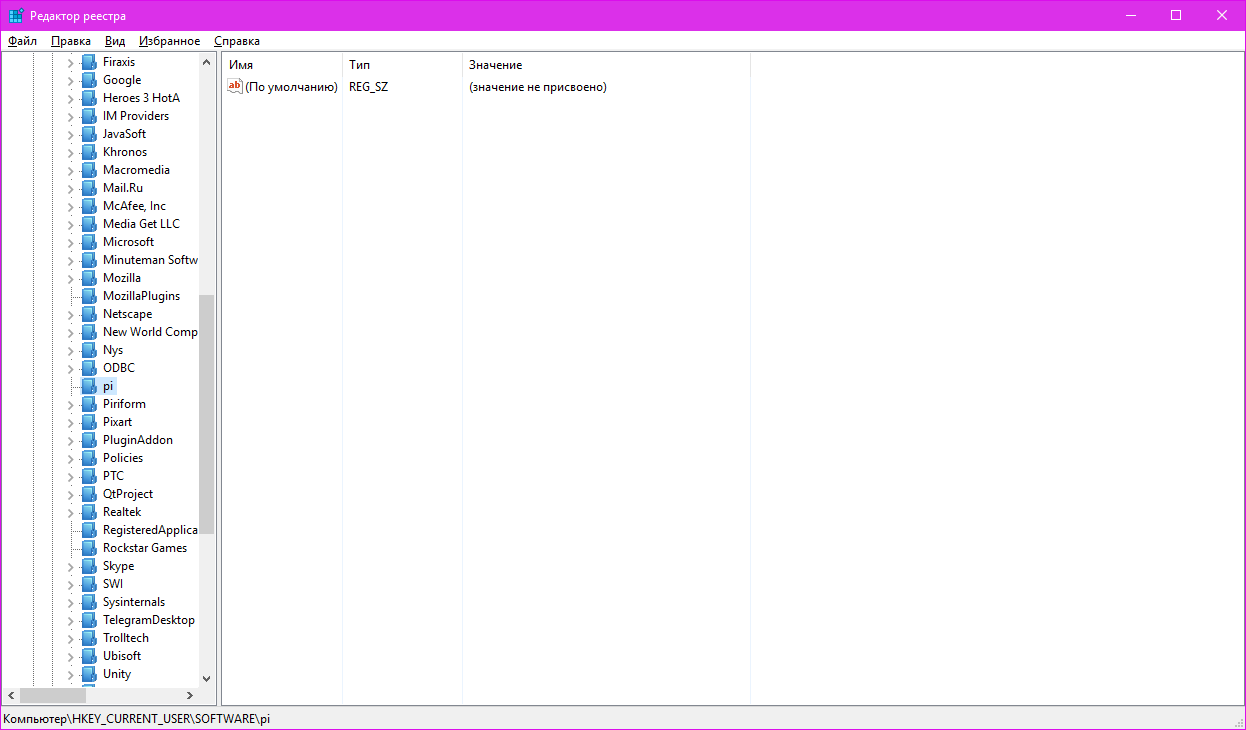
\_getch();

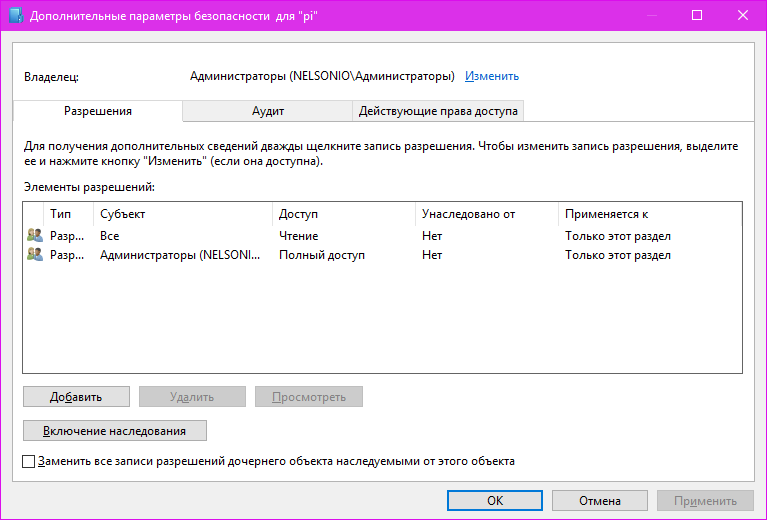
return 0;

}

**Результат работы программы:**







**Вывод:** Мы изучили методы работы с атрибутами безопасности в ОС Windows.