

Министерство науки и образования РФ  
Федеральное государственное автономное образовательное  
учреждение высшего профессионального образования  
«Санкт-Петербургский государственный электротехнический  
университет «ЛЭТИ» им. В. И. Ульянова (Ленина)»  
(СПбГЭТУ «ЛЭТИ»)

Кафедра вычислительной техники

Отчёт  
по лабораторной работе № 2  
на тему:  
“Безопасность в Windows”  
по дисциплине “Операционные системы”

Выполнил студент гр. 4306:  
Табаков А.В.  
Принял: Тимофеев А.В.

Санкт-Петербург  
2016

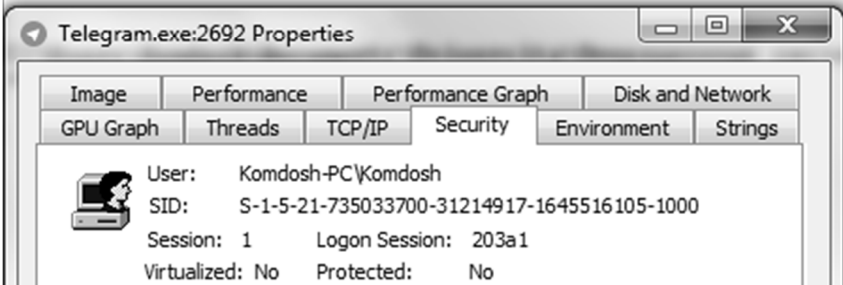
**Цель работы:** исследовать структуры данных Windows, используемые для обеспечения безопасности.

**Задание 2.1:** Определить идентификатор защиты SID текущего пользователя

```
C:\Users\Kondosh\Documents\University\Операционные системы\2 Лабораторная\PSTool
s>psgetsid komdosh

PsGetSid v1.44 - Translates SIDs to names and vice versa
Copyright (C) 1999-2008 Mark Russinovich
Sysinternals - www.sysinternals.com

SID for Kondosh-PC\komdosh:
S-1-5-21-735033700-31214917-1645516105-1000
```



**Вывод:** процесс имеет такие же права, как и пользователь, который его запустил

**Задание 2.2:** исследовать маркер доступа (Access token)

Был выбран процесс Telegram.exe с pid  $2692_{10} = A84_{16}$

```
0: kd> !process A84 1
*** ERROR: Module load completed but symbols could not be loaded for LiveKdD.SYS

Searching for Process with Cid == a84
PROCESS fffffa8008ed5060
  SessionId: 1 Cid: 0a84 Peb: 7efdf000 ParentCid: 069c
  DirBase: 1563a1000 ObjectTable: fffff8a004541820 HandleCount: 431.
  Image: Telegram.exe
  VadRoot fffffa800911a710 Vads 213 Clone 0 Private 7775. Modified 12651. Locked 0.
  DeviceMap fffff8a0020b6250
  Token fffff8a004541060
  ElapsedTime 00:09:42.246
  UserTime 00:00:00.608
  KernelTime 00:00:01.279
  QuotaPoolUsage[PagedPool] 267608
  QuotaPoolUsage[NonPagedPool] 26936
  Working Set Sizes (now,min,max) (12668, 50, 345) (50672KB, 200KB, 1380KB)
  PeakWorkingSetSize 12681
  VirtualSize 183 Mb
  PeakVirtualSize 197 Mb
  PageFaultCount 40530
  MemoryPriority BACKGROUND
  BasePriority 8
  CommitCharge 10112
```

Модификатор доступа (Access Token):

```
0: kd> dt _Token fffff8a004541060
nt!_TOKEN
+0x000 TokenSource      : _TOKEN_SOURCE
+0x010 TokenId          : _LUID
+0x018 AuthenticationId : _LUID
+0x020 ParentTokenId    : _LUID
+0x028 ExpirationTime   : _LARGE_INTEGER 0x7fffffff`ffffffff
+0x030 TokenLock        : 0xfffffa80`08ed4270 _ERESOURCE
+0x038 ModifiedId       : _LUID
+0x040 Privileges        : _SEP_TOKEN_PRIVILEGES
+0x058 AuditPolicy       : _SEP_AUDIT_POLICY
+0x074 SessionId        : 1
+0x078 UserAndGroupCount : 0x11
+0x07c RestrictedSidCount : 0
+0x080 VariableLength    : 0x324
+0x084 DynamicCharged    : 0x400
+0x088 DynamicAvailable : 0
+0x08c DefaultOwnerIndex : 5
+0x090 UserAndGroups     : 0xfffff8a0`04541368 _SID_AND_ATTRIBUTES
+0x098 RestrictedSids     : <null>
+0x0a0 PrimaryGroup      : 0xfffff8a0`04521410 Void
+0x0a8 DynamicPart       : 0xfffff8a0`04521410 -> 0x501
+0x0b0 DefaultDacl       : 0xfffff8a0`0452142c _ACL
+0x0b8 TokenType         : 1 < TokenPrimary >
+0x0bc ImpersonationLevel : 0 < SecurityAnonymous >
+0x0c0 TokenFlags         : 0x2000
+0x0c4 TokenInUse         : 0x1 ''
+0x0c8 IntegrityLevelIndex : 0x10
+0x0cc MandatoryPolicy    : 3
+0x0d0 LogonSession       : 0xfffff8a0`02644730 _SEP_LOGON_SESSION_REFERENCES
+0x0d8 OriginatingLogonSession : _LUID
+0x0e0 SidHash            : _SID_AND_ATTRIBUTES_HASH
+0x1f0 RestrictedSidHash  : _SID_AND_ATTRIBUTES_HASH
+0x300 pSecurityAttributes : 0xfffff8a0`04513c40 _AUTHZBASEP_SECURITY_ATTRIBUTES_INFORMATION
+0x308 VariablePart       : 0xfffff8a0`04541478
0: kd>
```

Чтение атрибутов процесса:

```
0: kd> dt _SID_AND_ATTRIBUTES 0xfffff8a004541368
nt!_SID_AND_ATTRIBUTES
+0x000 Sid              : 0xfffff8a0`04541478 Void
+0x008 Attributes       : 0
```

Чтение SID процесса:

```
0: kd> !sid 0xfffff8a004541478
SID is: S-1-5-21-735033700-31214917-1645516105-1000
0: kd> _
```

**Вывод:** Значение SID хранящееся в памяти совпадает с тем, что показал process explorer.

**Задание 2.3:** исследовать дескриптор защиты

Выведем краткую и полную информацию о googledrivesync.exe

```
0: kd> !process 109c 0
Searching for Process with Cid == 109c
PROCESS fffffa80091ab560
  SessionId: 1 Cid: 109c Peb: 7efdf000 ParentCid: 0d54
  DirBase: 1597ca000 ObjectTable: fffff8a003cdc080 HandleCount: 27.
  Image: googledrivesync.exe
```

```

0: kd> !process 109c 1
Searching for Process with Cid == 109c
PROCESS fffff80091ab560
  SessionId: 1 Cid: 109c Peb: 7efdf000 ParentCid: 0d54
  DirBase: 1597ca000 ObjectTable: fffff8a003cdc080 HandleCount: 27.
  Image: googledrivesync.exe
  VadRoot fffff800919dd70 Vads 58 Clone 0 Private 311. Modified 10052. Locked
0.
  DeviceMap fffff8a003063590
  Token fffff8a003dc69d0
  ElapsedTime 00:06:04.411
  UserTime 00:00:00.000
  KernelTime 00:00:00.000
  QuotaPoolUsage[PagedPool] 89912
  QuotaPoolUsage[NonPagedPool] 6720
  Working Set Sizes (now,min,max) <995, 50, 345> <3980KB, 200KB, 1380KB>
  PeakWorkingSetSize 2509
  VirtualSize 48 Mb
  PeakVirtualSize 50 Mb
  PageFaultCount 11584
  MemoryPriority BACKGROUND
  BasePriority 8
  CommitCharge 411

```

Определим заголовок объекта и читаем его

```

0: kd> !object fffff80091ab560
Object: fffff80091ab560 Type: (fffffa80060f8de0) Process
  ObjectHeader: fffff80091ab530 (new version)
  HandleCount: 1 PointerCount: 11
0: kd> dt _OBJECT_HEADER fffff80091ab530
nt!_OBJECT_HEADER
+0x000 PointerCount : 0n11
+0x008 HandleCount : 0n1
+0x008 NextToFree : 0x00000000'00000001 Void
+0x010 Lock : _EX_PUSH_LOCK
+0x018 TypeIndex : 0x7'
+0x019 TraceFlags : 0'
+0x01a InfoMask : 0x8'
+0x01b Flags : 0'
+0x020 ObjectCreateInfo : 0xfffffa80'0838bd80 _OBJECT_CREATE_INFORMATION
+0x020 QuotaBlockCharged : 0xfffffa80'0838bd80 Void
+0x028 SecurityDescriptor : 0xfffffa80'032b5a65 Void
+0x030 Body : _QUAD

```

Просмотрим дескриптор безопасности

```

0: kd> !sd fffff8a0032b5a65 & -8
->Revision: 0x1
->Sbz1 : 0x0
->Control : 0x8814
          SE_DACL_PRESENT
          SE_SACL_PRESENT
          SE_SACL_AUTO_INHERITED
          SE_SELF_RELATIVE
->Owner : S-1-5-32-544
->Group : S-1-5-21-735033700-31214917-1645516105-513
->Dacl :
->Dacl : ->AclRevision: 0x2
->Dacl : ->Sbz1 : 0x0
->Dacl : ->AclSize : 0x50
->Dacl : ->AceCount : 0x3
->Dacl : ->Sbz2 : 0x0
->Dacl : ->Ace[0]: ->AceType: ACCESS_ALLOWED_ACE_TYPE
->Dacl : ->Ace[0]: ->AceFlags: 0x0
->Dacl : ->Ace[0]: ->AceSize: 0x18
->Dacl : ->Ace[0]: ->Mask : 0x001fffff
->Dacl : ->Ace[0]: ->SID: S-1-5-32-544

->Dacl : ->Ace[1]: ->AceType: ACCESS_ALLOWED_ACE_TYPE
->Dacl : ->Ace[1]: ->AceFlags: 0x0
->Dacl : ->Ace[1]: ->AceSize: 0x14
->Dacl : ->Ace[1]: ->Mask : 0x001fffff
->Dacl : ->Ace[1]: ->SID: S-1-5-18

->Dacl : ->Ace[2]: ->AceType: ACCESS_ALLOWED_ACE_TYPE
->Dacl : ->Ace[2]: ->AceFlags: 0x0
->Dacl : ->Ace[2]: ->AceSize: 0x1c
->Dacl : ->Ace[2]: ->Mask : 0x00121411
->Dacl : ->Ace[2]: ->SID: S-1-5-5-0-562798

->Sacl :
->Sacl : ->AclRevision: 0x2
->Sacl : ->Sbz1 : 0x0
->Sacl : ->AclSize : 0x1c
->Sacl : ->AceCount : 0x1
->Sacl : ->Sbz2 : 0x0
->Sacl : ->Ace[0]: ->AceType: SYSTEM_MANDATORY_LABEL_ACE_TYPE
->Sacl : ->Ace[0]: ->AceFlags: 0x0
->Sacl : ->Ace[0]: ->AceSize: 0x14
->Sacl : ->Ace[0]: ->Mask : 0x00000003
->Sacl : ->Ace[0]: ->SID: S-1-16-12288

```

Дескриптор защиты содержит 3 элемента ACE «доступ разрешен» со следующими масками:

0x001fffff = 0000.0000.0001.1111.1111.1111.1111.1111  
0x00121411 = 0000.0000.0001.0010.0001.0100.0001.0001

А также один элемент SACL.

Константы-маски прав доступа из winnt.h:

```
// The following are masks for the predefined standard access types
#define DELETE (0x00010000L)
#define READ_CONTROL (0x00020000L)
#define WRITE_DAC (0x00040000L)
#define WRITE_OWNER (0x00080000L)
#define SYNCHRONIZE (0x00100000L)
#define STANDARD_RIGHTS_REQUIRED (0x000F0000L)
#define STANDARD_RIGHTS_READ (READ_CONTROL)
#define STANDARD_RIGHTS_WRITE (READ_CONTROL)
#define STANDARD_RIGHTS_EXECUTE (READ_CONTROL)
#define STANDARD_RIGHTS_ALL (0x001F0000L)
#define SPECIFIC_RIGHTS_ALL (0x0000FFFFL)
#define ACCESS_SYSTEM_SECURITY (0x01000000L)
#define MAXIMUM_ALLOWED (0x02000000L)
#define GENERIC_READ (0x80000000L)
#define GENERIC_WRITE (0x40000000L)
#define GENERIC_EXECUTE (0x20000000L)
#define GENERIC_ALL (0x10000000L)
```

Таким образом, рассматриваемый googledrivesync.exe имеет все стандартные права:

DELETE - Delete access

READ\_CONTROL - Read access to the owner, group, and DACL of the security descriptor.

WRITE\_DAC - Write access to the DACL.

WRITE\_OWNER - Write access to owner

SYNCHRONIZE - Synchronize access

И все object-specific

Вывод: дескриптор защиты имеет сложную структуру, с помощью которой производится тонкая настройка прав доступа.

Причём у 0 и 1 элементов следующие права:

0x001fffff = 0000.0000.0001.1111.1111.1111.1111.1111

DELETE  
READ\_CONTROL  
WRITE\_DAC  
WRITE\_OWNER  
SYNCHRONIZE  
STANDARD\_RIGHTS\_REQUIRED  
STANDARD\_RIGHTS\_READ  
STANDARD\_RIGHTS\_WRITE  
STANDARD\_RIGHTS\_EXECUTE  
STANDARD\_RIGHTS\_ALL  
SPECIFIC\_RIGHTS\_ALL

А у элемента 2 их меньше:

0x00121411 = 0000.0000.0001.0010.0001.0100.0001.0001

READ\_CONTROL  
SYNCHRONIZE  
STANDARD\_RIGHTS\_READ  
STANDARD\_RIGHTS\_WRITE  
STANDARD\_RIGHTS\_EXECUTE