Что полезного в разборе дампов для .NET-разработчиков?

Alexander Rakhmanov

KASPERSKY lab

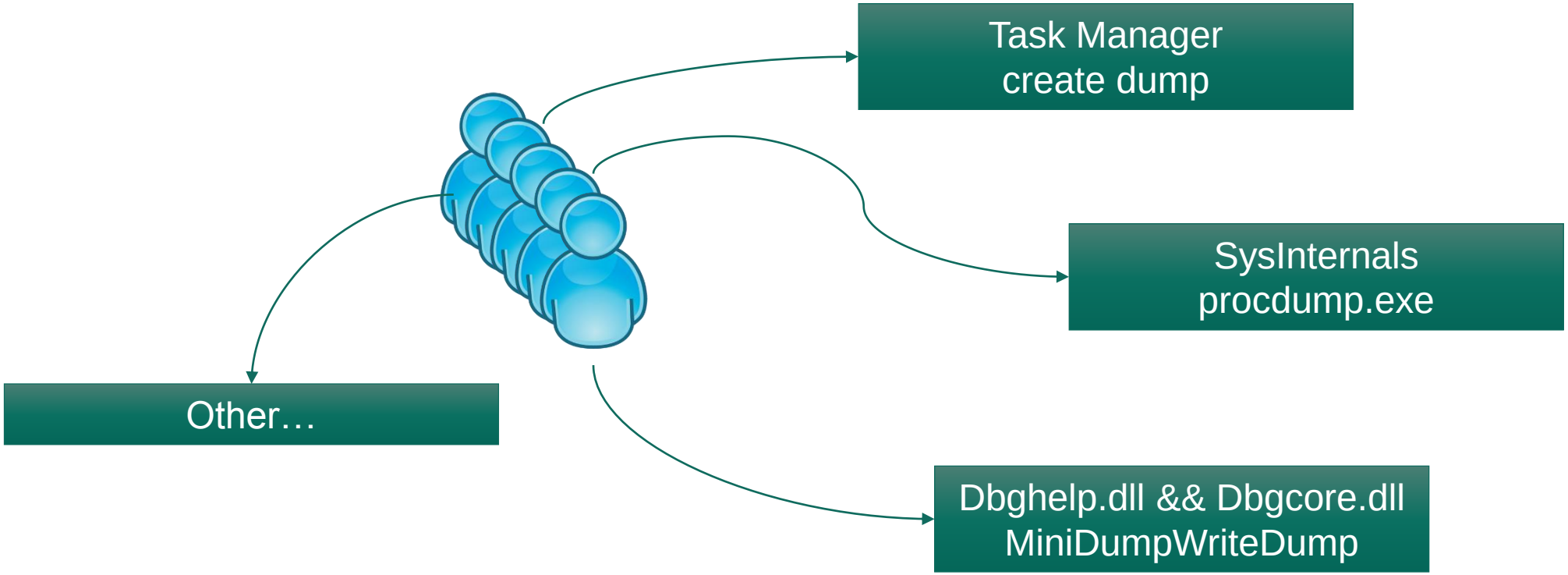
О чем поговорим?

- Как обработка дампов встроена в процесс поддержки пользователей?
- Какие виды проблем можно диагностировать при помощи дампов?
- Интересные примеры из практики

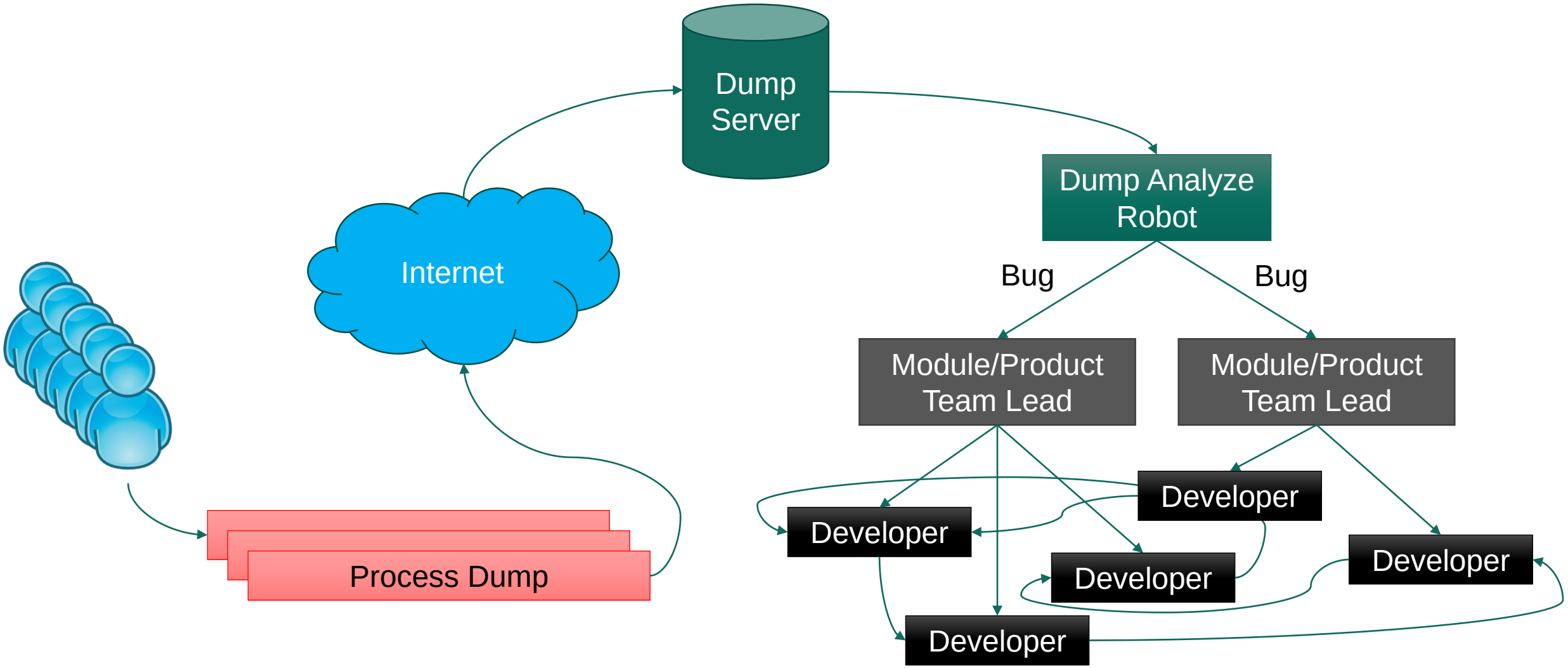
Почему про дампы?

- Стандартный для многих C++ разработчиков подход все еще не распространен
- По многим докладам создается впечатление «ненужности» анализа дампов для рядового разработчика
- Хотим поделиться опытом наших падений ☺

От куда у нас берутся дампы?



Процесс: от пользователя до разработчика



Этапы анализа

- Запись дампа на машине пользователя
- Отправка на дамп сервер
- Анализ на стороне дамп сервера
  - группировка
  - удаление дубликатов
  - запись статистики
- Создание дефекта при превышении порога
- Автоматическое распределение на команду
- Анализ и исправление со стороны разработчика
- Внутреннее тестирование и закрытие дефекта

- Также дефект может быть открыт повторно автоматически, если воспроизвелся на более новой версии

Какие могут быть триггеры для записи?

- Ручной дамп
- Автоматически на Unhandled Exception
- First Chance обработчики
- Asserts
- Watchdogs

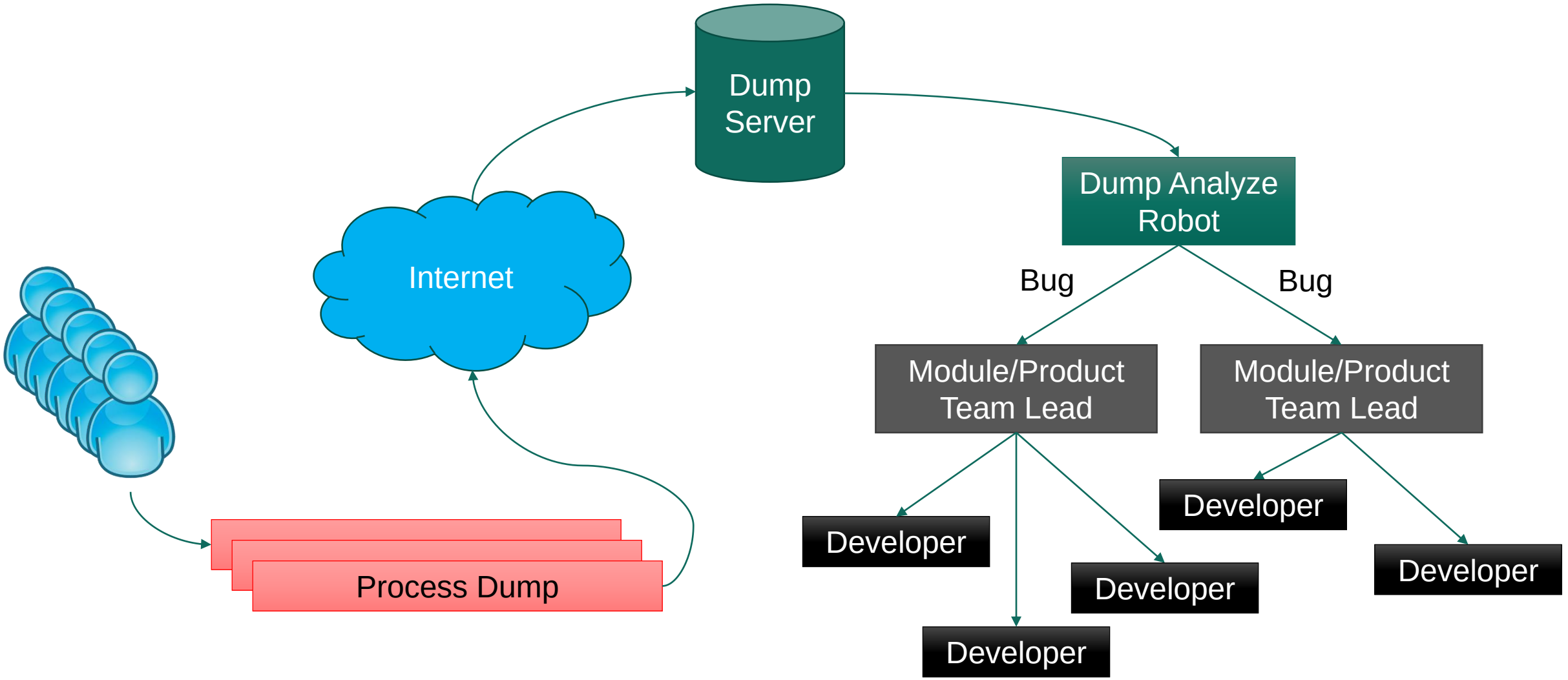
Как можно себе помочь в дальнейшем при анализе дампа?

- Хранение последних килобайт трассировок прямо в памяти
- Автоматическое создание дампов связанных процессов
- Запись информации об окружении прямо в дампе или связанных файлах
- Запись дампа наиболее близко к месту падения
  - избегать записи дампа в catch блоке

Помощники при анализе дампов

- Скрипты для автоматической подгрузки sos.dll, вывод исключения и пр. на запуске WinDBG
- Скрипты для анализа
  - поиск и вывод диагностики приложения
  - сбор статистики по объектам в памяти
- ClrMD: Microsoft.Diagnostics.Runtime
  - разработка собственных анализаторов на C#

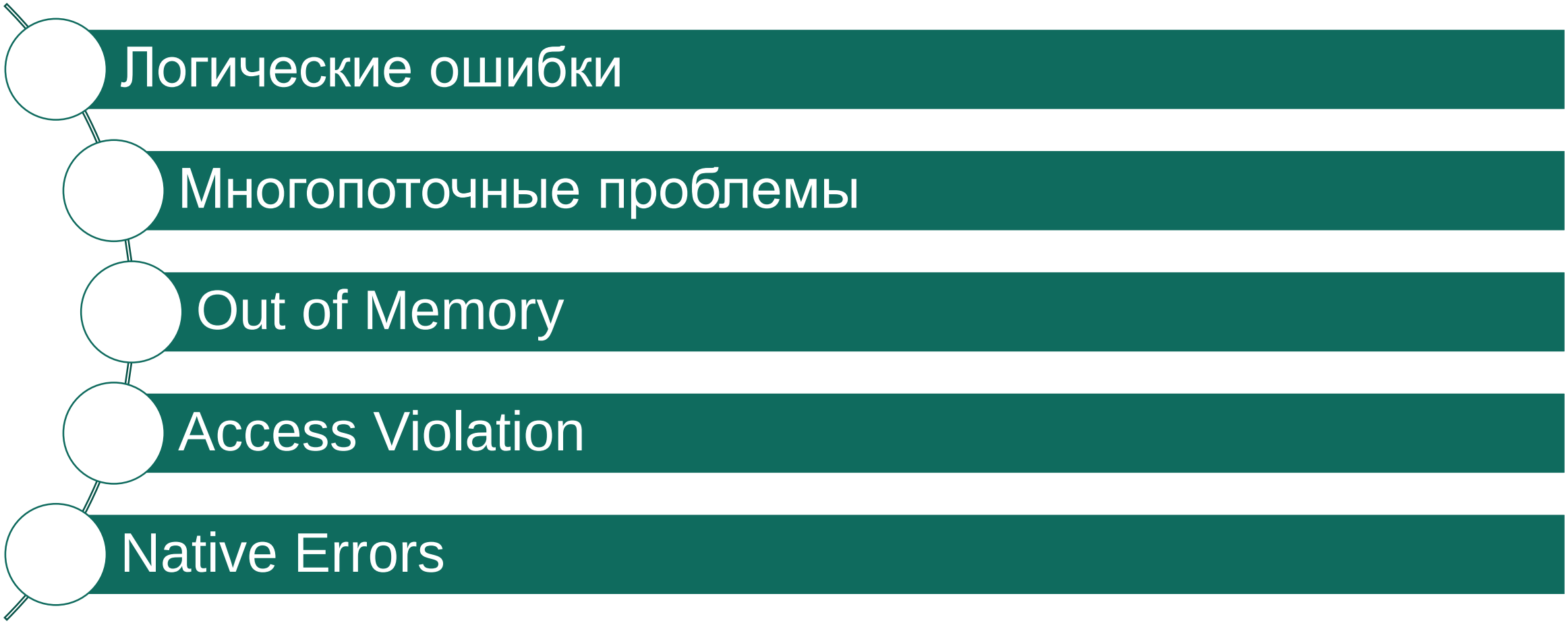
Еще раз взглянем на общую картину



Приступим к анализу?

KASPERSKY lab

Виды ошибок


Логические ошибки
Многопоточные проблемы
Out of Memory
Access Violation
Native Errors

Логические ошибки

Средства диагностики

- Трассировки
- Отладчик
- Анализ дампов
  - Visual Studio
  - WinDBG

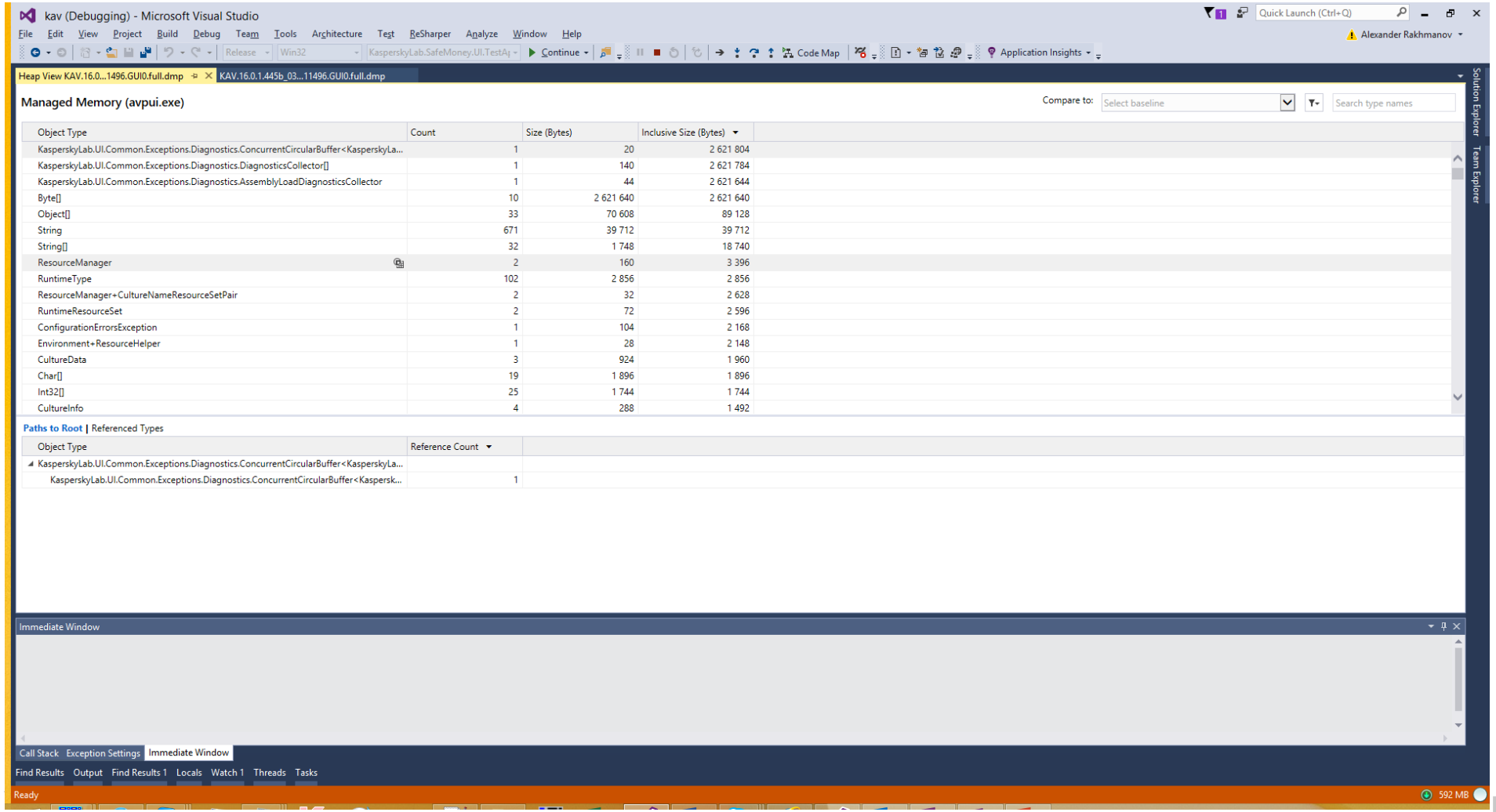
Логические ошибки

Плюсы и минусы инструментов для анализа дампов

- Visual Studio
  - ✓ Удобный просмотр объектов в привычном виде
  - ✓ Managed Code Evaluation
  - ✓ Быстрый просмотр Managed Heap
- WinDBG
  - ✓ Скрипты
  - ✓ Удобный анализ при помощи команд
  - ✓ Доступ к объектам на низком уровне
  - ✓ Одновременный анализ managed & native объектов
  - ✓ Отображение IL, Native & Source Code
  - ✓ SOS и SOSEX

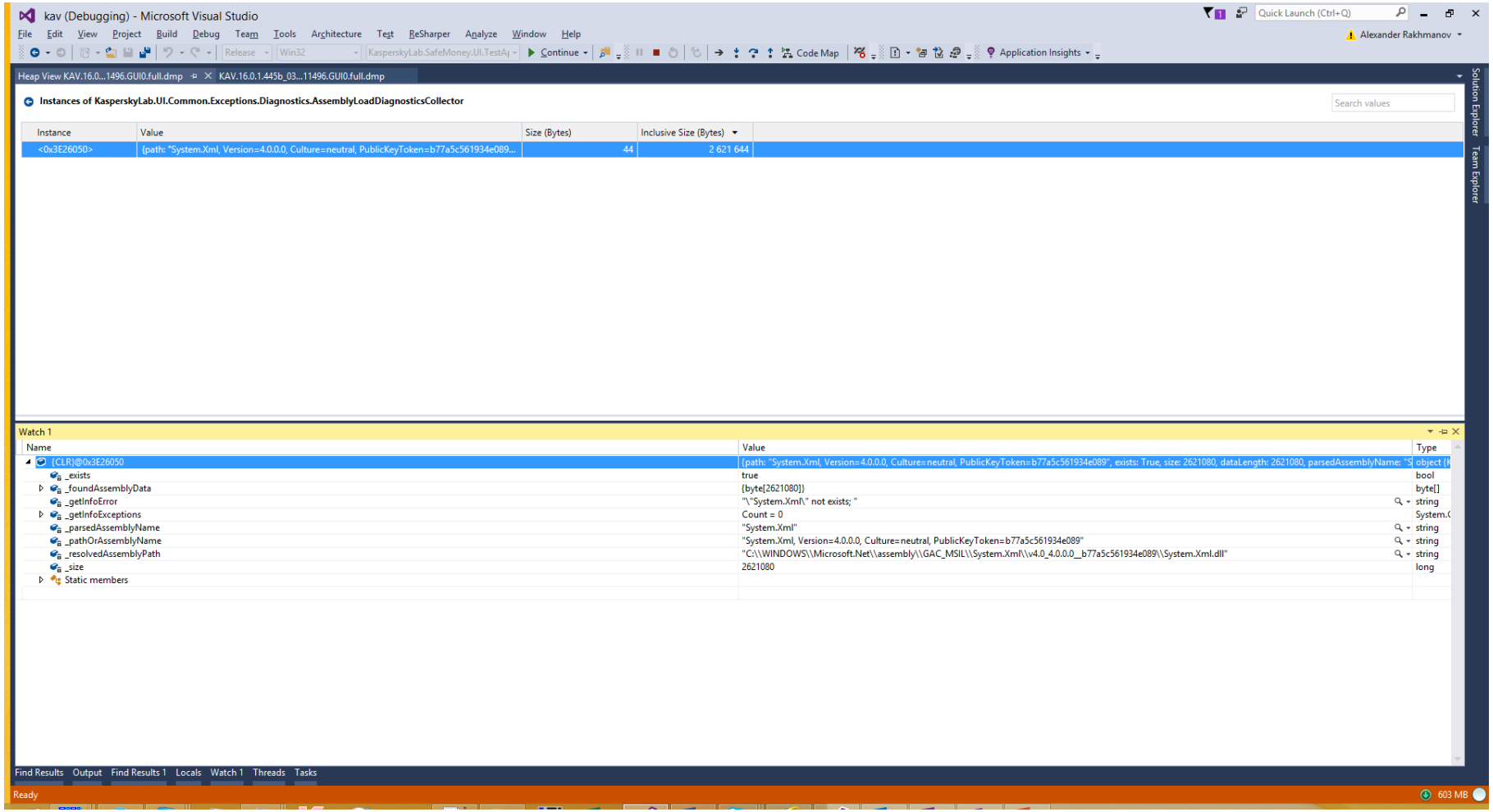
Логические ошибки

Пример 1. Visual Studio. Отображение managed heap



Логические ошибки

Пример 2. Visual Studio. Отображение managed объектов



Логические ошибки

Пример 3. WinDBG. Отображение IL

```
0:016> !dumpil 00dc90c0
ilAddr = 06b06a2e
IL_0000: ldarg.0
IL_0001: ldc.i4.7
IL_0002: ldarg.1
IL_0003: call
System.Globalization.NumberFormatInfo::GetInstance
IL_0008: call System.Number::ParseInt32
IL_000d: ret
```

Логические ошибки

Пример 4. WinDBG. Отображение ассемблерного кода

```
0:016> !u 048e0210
Normal JIT generated code
System.Int32.Equals(System.Object)
Begin 048e0210, size 47
>>> 048e0210 57              push    edi
048e0211 56              push    esi
048e0212 8bf9            mov     edi,ecx
048e0214 8bf2            mov     esi,edx
048e0216 85f6            test    esi,esi
048e0218 7410            je      048e022a
048e021a 813e2093dc00    cmp     dword ptr [esi],0DC9320h
048e0220 b800000000      mov     eax,0
048e0225 0f44c6          cmove   eax,esi
048e0228 eb02            jmp     048e022c
048e022a 8bc6            mov     eax,esi
048e022c 85c0            test    eax,eax
048e022e 7505            jne     048e0235
048e0230 33c0            xor     eax,eax
048e0232 5e              pop     esi
```

Логические ошибки

Пример 5. WinDBG. Объекты «на стэке» и в куче

```
0a34f010       18         216 KasperskyLab.UI.Clr.ClrScopedPtr<eka::objptr_t<eka::IObject> >
0abe08e8       12         384 System.Linq.Enumerable+WhereEnumerableIterator`1[[System.Globalization.CultureInfo,
mscorlib]]
0abe0470       12         384 System.Func`2[[System.Globalization.CultureInfo,
mscorlib],[System.Collections.Generic.List`1[[Nemerle.Builtins.Function`2[[System.Int32, mscorlib],[System.Boolean,
mscorlib]], Nemerle]], mscorlib]]
0492fa88       12         384 System.Func`2[[System.Globalization.CultureInfo, mscorlib],[System.Boolean, mscorlib]]
0492dbc4        2         392 System.Globalization.DateTimeFormatInfo
0abe09e8       12         432
System.Linq.Enumerable+WhereSelectEnumerableIterator`2[[System.Globalization.CultureInfo,
mscorlib],[System.Collections.Generic.List`1[[Nemerle.Builtins.Function`2[[System.Int32, mscorlib],[System.Boolean,
mscorlib]], Nemerle]], mscorlib]]
0ad20608        6         552 System.Globalization.TimeSpanParse+TimeSpanToken[]
0492b900        5         660 System.Globalization.NumberFormatInfo
```

Логические ошибки

Пример 6. WinDBG. Состояние method table

```
0:016> !DumpMT -md 00dc9320
EEClass:      00e10618
Module:       009e24b0
Name:         System.Int32
mdToken:      020000ec
File:         C:\Windows\Microsoft.Net\assembly\GAC_32\mscorlib\v4.0_4.0.0.0__b77a5c561934e089\mscorlib.dll
BaseSize:         0xc
ComponentSize:    0x0
Slots in VTable: 56
Number of IFaces in IFaceMap: 5
--------------------------------------
MethodDesc Table
   Entry MethodDesc      JIT Name
00c739b8   00dc9060     NONE System.Int32.ToString()
00c739c8   00dc9024     NONE System.Int32.Equals(System.Object)
00c739d8   00dc904c     NONE System.Int32.GetHashCode()
009ec02d   00b0575c      JIT System.Object.Finalize()
```

Логические ошибки

Пример 7. Что делать, если ошибка в чужом коде?

Выгрузка модулей

→ Генерация символов

→ Local Symbol Server

→ Отладка в Visual Studio

Многопоточные проблемы

Варианты?

- WinDBG

Многопоточные проблемы

Состояния потоков

```
0:016> !threads
ThreadCount:      6
UnstartedThread:  0
BackgroundThread: 5
PendingThread:    0
DeadThread:       0
Hosted Runtime:   no
                                PreEmptive   GC Alloc           Lock
      ID  OSID ThreadOBJ    State GC         Context       Domain   Count APT Exception
  14    2   ddc 003bcf00      b220 Enabled  04f68e68:04f6ae34 003af4d8     0 MTA (Finalizer)
  16    4   2f0 03a3a350      7020 Enabled  054d0dd8:054d14fc 003af4d8     1 STA
System.ArgumentOutOfRangeException (05477d10)
  18    3   2fc 03a5ea50   1009220 Enabled  04f6ce7c:04f6ee34 003af4d8     0 MTA (Threadpool Worker)
  19    1   964 004038f8   1009220 Enabled  04f6ae70:04f6ce34 003af4d8     0 MTA (Threadpool Worker)
  21    5   ff4 0b19ff48   1009220 Enabled  00000000:00000000 003af4d8     0 MTA (Threadpool Worker)
  22    6   ca4 03a9f1c0   1009220 Enabled  00000000:00000000 003af4d8     0 MTA (Threadpool Worker)
```

Многопоточные проблемы

Что внутри lock? А внутри Monitor? А дальше? ☺

```
0:022> !clrstack
OS Thread Id: 0xf84 (22)
Child SP IP       Call Site
0a40f018 7c90e514 [GCFrame:
0a40f018]
0a40f10c 7c90e514
[HelperMethodFrame_1OBJ: 0a40f10c]
System.Threading.Monitor.ObjWait(Boo
lean, Int32, System.Object)
0a40f168 08188ae2
System.Threading.Monitor.Wait(System
.Object, Int32, Boolean)
0a40f178 08188559
System.Threading.ManualResetEventS
lim.Wait(Int32,
System.Threading.CancellationToken)
02 0a40ede0 792dedaa kernel32!WaitForMultipleObjectsEx+0x12c
03 0a40ee44 792debd8 clr!WaitForMultipleObjectsEx_SO_TOLERANT+0x
04 0a40ee64 792dea17 clr!Thread::DoAppropriateAptStateWait+0x4d
05 0a40eef8 792dead8 clr!Thread::DoAppropriateWaitWorker+0x17d
06 0a40ef64 792deb57 clr!Thread::DoAppropriateWait+0x60
07 0a40efb8 7916173e clr!CLREvent::WaitEx+0x106
08 0a40efcc 792ad6a4 clr!CLREvent::Wait+0x19
09 0a40efe0 792ad6c2 clr!Thread::Wait+0x1d
0a 0a40eff4 792ad4cd clr!Thread::Block+0x1a
0b 0a40f0a0 792ad53a clr!SyncBlock::Wait+0x169
0c 0a40f0b4 792ad62c clr!ObjHeader::Wait+0x2c
0d 0a40f15c 08188ae2 clr!ObjectNative::WaitTimeout+0x147
```

Многопоточные проблемы

Пример deadlock в инициализации WPF

clr!ArgIteratorTemplate<ArgIteratorBase>::ForceSigWalk
0b70e338 6e3027e6 clr!Thread::DoSyncContextWait+0xb4, calling clr!MethodDescCallSite::CallTargetWorker
0b70e418 6e27a520 clr!Thread::DoAppropriateWaitWorker+0x100, calling clr!Thread::DoSyncContextWait
0b70e4a8 6e3ba2e6 clr!Thread::DoAppropriateWait+0x64, calling clr!Thread::DoAppropriateWaitWorker
0b70e4fc 6e3091b6 clr!AcquireSafeHandle+0x33, calling clr!_EH_epilog3
0b70e514 6e309334 clr!WaitHandleNative::CorWaitOneNative+0x16b, calling clr!Thread::DoAppropriateWait
0b70e58c 6e309269 clr!WaitHandleNative::CorWaitOneNative+0x4c, calling clr!LazyMachStateCaptureState
0b70e5b0 6e23a510 clr!ETWTraceStartup::~ETWTraceStartup+0x10, calling clr!ETWTraceStartup::StartupTraceEvent
0b70e5c0 6e23b075 clr!PreStubWorker+0x13e, calling ntdll!RtlSetLastWin32Error
0b70e61c 08f63c9c (MethodDesc 07fa8204 +0x1c
System.Threading.WaitHandle.InternalWaitOne(System.Runtime.InteropServices.SafeHandle, Int64, Boolean, Boolean)), calling
clr!WaitHandleNative::CorWaitOneNative
0b70e630 08f63c1d (MethodDesc 07fa81dc +0x25 System.Threading.WaitHandle.WaitOne(Int32, Boolean)), calling
(MethodDesc 07fa8204 +0 System.Threading.WaitHandle.InternalWaitOne(System.Runtime.InteropServices.SafeHandle, Int64,
Boolean, Boolean))
0b70e64c 08f63be0 (MethodDesc 07fa81ec +0x10 System.Threading.WaitHandle.WaitOne())
0b70e658 04007291 (MethodDesc 0407a364 +0xb1 System.Windows.Input.PenThreadWorker.WorkerGetTabletsInfo())
0b70e68c 04006994 (MethodDesc 0a21e678 +0x44 System.Windows.Input.StylusLogic.GetDeviceCount()), calling
(MethodDesc 0407a364 +0 System.Windows.Input.PenThreadWorker.WorkerGetTabletsInfo())
0b70e69c 063fd0f7 (MethodDesc 0a21e344 +0x47 System.Windows.Input.StylusLogic.get_TabletDevices()), calling
(MethodDesc 0a21e678 +0 System.Windows.Input.StylusLogic.GetDeviceCount())
0b70e6ac 063fd1aa (MethodDesc 0a21e3b0 +0x92
System.Windows.Input.StylusLogic.RegisterHwndForInput(System.Windows.Input.InputManager,
System.Windows.PresentationSource)), calling (MethodDesc 0a21e344 +0
System.Windows.Input.StylusLogic.get_TabletDevices())

Многопоточные проблемы

Не thread-safe jitter

```
0:018> k
 # ChildEBP RetAddr
00 09bddf80 7917b03c clr!_EH_prolog3_catch+0x11
01 09bddfa0 09ee80d9 clr!PreStubWorker+0xc
WARNING: Frame IP not in any known module. Following frames may be
wrong.
02 09bde068 7914219b 0x9ee80d9
03 09bde094 7917a7aa clr!CallDescrWorker+0x33
04 09bde110 7917a94c clr!CallDescrWorkerWithHandler+0x8e
05 09bde270 7917a981 clr!MethodDesc::CallDescr+0x194
06 09bde28c 7917a9a1 clr!MethodDesc::CallTargetWorker+0x21
07 09bde2a4 791c7c8e clr!MethodDescCallSite::Call+0x1c
08 09bde4c0 791c7e1b clr!InvokeConstructorHelper+0x4b2
09 09bde580 0987b33e
clr!RuntimeMethodHandle::InvokeConstructor+0x161
```

Многопоточные проблемы

SOSEX extension

```
0:000> .load sosex
This dump has no SOSEX heap index.
The heap index makes searching for references and roots
much faster.
To create a heap index, run !bhi
0:000> !mlocks
Examining SyncBlocks...
Scanning for ReaderWriterLock instances...
Scanning for holders of ReaderWriterLock locks...
Scanning for ReaderWriterLockSlim instances...
Scanning for holders of ReaderWriterLockSlim locks...
Examining CriticalSections...

ClrThread  DbgThread  OsThread    LockType    Lock
LockLevel
-----------------------------------------------------------------------
0xa        21         0xe60       CritSect    09dd1100
0xd        23         0xf88       SyncBlock   001d32e4
```

Out of Memory

Варианты?

- WinDBG
- Scitech Memory Profiler

Out of Memory

Плюсы и минусы инструментов

- WinDBG
  - ✓ Быстрый поиск по памяти
  - ✓ Сбор статистики, табличное сравнение
  - ✓ Возможность работы с native объектами
  - ✓ Вывод графа доступности объекта
- Scitech Memory Profiler
  - ✓ Удобный просмотр всего графа объектов
  - ✓ Просмотр свойств отдельных объектов в процессе анализа

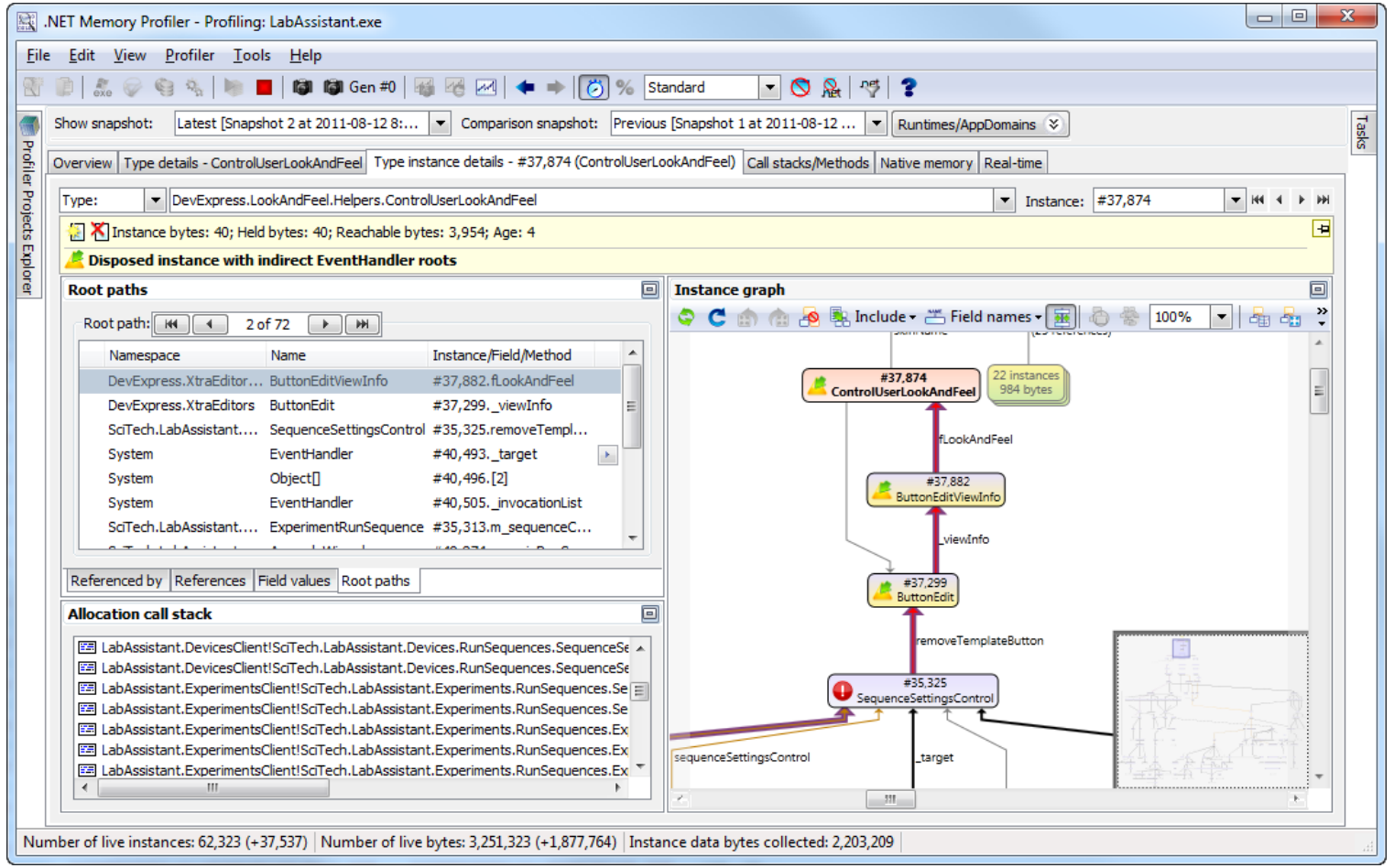
Out of Memory

Плюсы и минусы инструментов

- WinDBG
  - ✓ Быстрый поиск по памяти
  - ✓ Сбор статистики, табличное сравнение
  - ✓ Возможность работы с native объектами
  - ✓ Вывод графа доступности объекта
- Scitech Memory Profiler
  - ✓ Удобный просмотр всего графа объектов
  - ✓ Просмотр свойств отдельных объектов в процессе анализа

Out of Memory

Пример. <TODO>



Access Violation

Пример 1. Повреждения сборок

Exception type: System.BadImageFormatException
Message: Could not load file or assembly 'System.Xml, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089' or one of its dependencies. The module was expected to contain an assembly manifest.
InnerException: <none>
StackTrace (generated):
SP IP Function
00000000 00000001 UNKNOWN!System.Configuration.ExceptionUtil.WrapAsConfigException(System.String, System.Exception, System.String, Int32)+0x2
036FEDAC 08B33087 UNKNOWN!System.Configuration.BaseConfigurationRecord.Init(System.Configuration.Internal.IInternalConfigRoot, System.Configuration.BaseConfigurationRecord, System.String, System.String)+0xbd7
036FEF54 08B31C85 UNKNOWN!System.Configuration.RuntimeConfigurationRecord.Create(System.Configuration.Internal.InternalConfigRoot, System.Configuration.Internal.IInternalConfigRecord, System.String)+0x45
036FEF68 08B360F6 UNKNOWN!System.Configuration.Internal.InternalConfigRoot.GetConfigRecord(System.String)+0x27e
036FEFB8 08B35C87 UNKNOWN!System.Configuration.ClientConfigurationSystem.EnsureInit(System.String)+0xd7

Access Violation

Пример 2. Повреждения кода

```
707a0cc0 8d4d90          lea     ecx,[ebp-70h]
707a0cc3 e8771deaff      call    clr!HelperMethodFrame::Push (70642a3f)
707a0cc8 8b4da0          mov     ecx,dword ptr [ebp-60h]
707a0ccb 895dfc          mov     dword ptr [ebp-4],ebx
707a0cce 803d0ca3c77000  cmp     byte ptr [clr!g_StackProbingEnabled (70c7a30c)],0
707a0cd5 0f8586352a00    jne     clr!MngdNativeArrayMarshaler::ClearNative+0x46 (70a44261)
707a0cdb c645fc01        mov     byte ptr [ebp-4],1
707a0cdf 391e            cmp     dword ptr [esi],ebx
707a0ce1 0f85c7000000    jne     clr!MngdNativeArrayMarshaler::ClearNative+0x64 (707a0dae)
707a0ce7 885de0          mov     byte ptr [ebp-20h],bl
707a0cea 885dfc          mov     byte ptr [ebp-4],bl
707a0ced 803d0ca3c77000  cmp     byte ptr [clr!g_StackProbingEnabled (70c7a30c)],0
707a0cf4 0f8586352a00    jne     clr!MngdNativeArrayMarshaler::ClearNative+0x87 (70a44280)
707a0cfa 834dfcff        or      dword ptr [ebp-4],0FFFFFFFFh
707a0cfe 8d4d90          lea     ecx,[ebp-70h]
707a0d01 e8621deaff      call    clr!HelperMethodFrame::Pop (70642a68)
707a0d06 8d4da8          lea     ecx,[ebp-58h]
707a0d09 e86d18eadf      call    5064257b
707a0d0e 85c0            test    eax,eax
```

Access Violation

Пример 3. C++/CLI. KeepAlive

```
    template <typename T>
    struct PragueConverter<T^, cSerializable *>
    {
        static void Set(T^ value, StructHolder^ holder, size_t offset, size_t size)
        {
            if (value != nullptr && holder != nullptr)
            {
                cSerObj<cSerializable>* serObj = GetPointer<cSerObj<cSerializable> >(holder,
offset, size);
                serObj->assign(static_cast<cSerializable*>(value->Holder->Value), false /*
does not copy*/);
                GC::KeepAlive(value);
                GC::KeepAlive(this);
            }
        }
    };
```

Native Errors

Пример 1. Работа с native объектами

```
0:008> dt 047ff5f4 eka::CheckResultFailedException
avpui!eka::CheckResultFailedException
   +0x000 __VFN_table : 0x6d5b20f4
   +0x004 _Mywhat          : (null)
   +0x008 _Mydofree        : 0
   +0x00c m_fileName       : 0x6d5b33c0
"D:\207\7937\Sources\include\safe_banking/ui_host.h"
   +0x010 m_line           : 0n41
   +0x014 m_id             : 0n0
   +0x018 m_description    : eka::types::basic_string_t<unsigned
short,eka::char_traits<unsigned short>,eka::Allocator<unsigned short> >
   +0x038 m_nestedException : (null)
   +0x03c m_result         : 0n-2147417856
```

Native Errors

Пример 2. WPF OpenAdapter

```
0e 0bf0d7a8 0c32d711 atiumdva!mmdllInitUMDFuncs+0x3a8
0f 0bf0d984 0c2f1268 atiumdag!OpenAdapter+0x3ab01
10 0bf0d9a0 0c2f2c4f atiumdag!DllMain+0x208
*** ERROR: Symbol file could not be found.  Defaulted to export symbols for atiu9pag.dll -
11 0bf0d9c8 0b2c3133 atiumdag!OpenAdapter+0x3f
12 0bf0da04 0b2c34a4 atiu9pag!OpenAdapter+0x1aad
*** WARNING: Unable to verify checksum for aticfx32.dll
*** ERROR: Symbol file could not be found.  Defaulted to export symbols for aticfx32.dll -
13 0bf0da24 5ce40a8c atiu9pag!XopOpenAdapter9+0x5c
14 0bf0dadc 5ce4343e aticfx32!OpenAdapter+0x11ac
15 0bf0db08 5ce433b0 aticfx32!OpenAdapter+0x3b5e
16 0bf0db20 5ce408dc aticfx32!OpenAdapter+0x3ad0
17 0bf0e264 5ce3f930 aticfx32!OpenAdapter+0xffc
18 0bf0e278 669cb634 aticfx32!OpenAdapter+0x50
19 0bf0e5d8 669cb431 d3d9!CreateDeviceLHDDI+0x2dc
```

Native Errors

Пример 3. <TODO>

```
MANAGED_OBJECT: !dumpobj afb3984
Name:        System.String
MethodTable: 02ebfbd0
EEClass:     02ec8924
Size:        218(0xda) bytes
File:
C:\WINDOWS\Microsoft.Net\assembly\GAC_32\mscorlib\
v4.0_4.0.0.0__b77a5c561934e089\mscorlib.dll
String:      Attempted to read or write protected memory.
This is often an indication that other memory is corrupt.
```

```
LAST_CONTROL_TRANSFER:  from 651c2b3d to
651a2e20

STACK_TEXT:
080ae930 651c2b3d 0000001c 02ef4160 02f2631c
params!pr_updatable_settings::cLicAddInfoDate::operator
=+0x9
080ae958 651d6c9d 02f266ec 02ef3d9c 080ae980
params!cVector<cFileDownloadControlItem,cCtrFactory<c
FileDownloadControlItem> >::assign+0x5e
```

Интересные примеры из практики

KASPERSKY lab

Пример 1. Зависание в Stylus Input

21de668 1001f8dd (MethodDesc 1044221c +0x9d System.Windows.Input.PenThreadWorker..ctor()), calling 0ff6c9a8
121de6a0 10db8cd3 (MethodDesc 10441fec +0xf3
System.Windows.Input.PenThreadPool.GetPenThreadForPenContextHelper(System.Windows.Input.PenContext)), calling (MethodDesc 1044221c +0
System.Windows.Input.PenThreadWorker..ctor())
121de6c0 10db8a35 (MethodDesc 10441fd0 +0x6d
System.Windows.Input.PenThreadPool.GetPenThreadForPenContext(System.Windows.Input.PenContext)), calling (MethodDesc 10441fec +0
System.Windows.Input.PenThreadPool.GetPenThreadForPenContextHelper(System.Windows.Input.PenContext))
121de6d0 1001f504 (MethodDesc 10441c50 +0x34 System.Windows.Input.TabletDeviceCollection.UpdateTablets()), calling (MethodDesc 10441fd0 +0
System.Windows.Input.PenThreadPool.GetPenThreadForPenContext(System.Windows.Input.PenContext))
121de6f0 7313b07c clr!PreStubWorker+0x145, calling clr!_EH_epilog3
121de6f4 73122a0c clr!ThePreStub+0x16, calling clr!PreStubWorker
121de724 1001f26e (MethodDesc 10441c3c +0x4e System.Windows.Input.TabletDeviceCollection..ctor()), calling (MethodDesc 10441c50 +0
System.Windows.Input.TabletDeviceCollection.UpdateTablets())
121de734 104f16c1 (MethodDesc 10d1c8f4 +0x21 System.Windows.Input.StylusLogic.get_TabletDevices()), calling (MethodDesc 10441c3c +0
System.Windows.Input.TabletDeviceCollection..ctor())
121de744 104f1b4c (MethodDesc 10d1c960 +0x94 System.Windows.Input.StylusLogic.RegisterHwndForInput(System.Windows.Input.InputManager,
System.Windows.PresentationSource)), calling (MethodDesc 10d1c8f4 +0 System.Windows.Input.StylusLogic.get_TabletDevices())
121de784 1001f15b (MethodDesc 10441aec +0xcb System.Windows.Interop.HwndStylusInputProvider..ctor(System.Windows.Interop.HwndSource)),
calling (MethodDesc 10d1c960 +0 System.Windows.Input.StylusLogic.RegisterHwndForInput(System.Windows.Input.InputManager,
System.Windows.PresentationSource))
121de7d8 10d37a4d (MethodDesc 10007614 +0x38d
System.Windows.Interop.HwndSource.Initialize(System.Windows.Interop.HwndSourceParameters)), calling (MethodDesc 10441aec +0
System.Windows.Interop.HwndStylusInputProvider..ctor(System.Windows.Interop.HwndSource))

Пример 2. DateTime.Now exception

```
HResult: 80131502
0:016> !pe
Exception object: 05477d10
Exception type:   System.ArgumentOutOfRangeException
Message:          Year must be between 1 and 9999.
InnerException:   <none>
StackTrace (generated):
    SP       IP       Function
    095DD9D4 0B880E5F
UNKNOWN!System.DateTime.IsLeapYear(Int32)+0x9f
```

Пример 3. <TODO>

LET’S TALK?

Kaspersky Lab HQ
39A/3 Leningradskoe Shosse
Moscow, 125212, Russian Federation
Tel: +7 (495) 797-8700
www.kaspersky.com

KASPERSKY lab

WHO WE ARE,
WHAT WE DO,
WHAT MATTERS TO US

Secondary text area

KASPERSKY lab

We believe that everyone – from home computer users through to large corporations and governments – should be able to protect what matters to them most. Whether it's privacy, family, finances, customers, business success or critical infrastructure, we've made it our mission to secure it all.

Eugene Kaspersky,

chairman and CEO, Kaspersky Lab

TEXT SLIDE

WHAT BRINGS US TOGETHER

We believe that everyone – from home computer users through to large corporations and governments – should be able to protect what matters to them most. Whether it's privacy, family, finances, customers, business success or critical infrastructure, we've made it our mission to secure it all.

TEXT SLIDE

We believe that everyone – from home computer users through to large corporations and governments – should be able to protect what matters to them most. Whether it’s privacy, family, finances, customers, business success or critical infrastructure, we’ve made it our mission to secure it all.

BULLET SLIDE

- Образец текста
  - Второй уровень
    - Третий уровень
      - Четвертый уровень
        - Пятый уровень

TEXT SLIDE

WHAT BRINGS US TOGETHER

We believe that everyone – from home computer users through to large corporations and governments – should be able to protect what matters to them most. Whether it’s privacy, family, finances, customers, business success or critical infrastructure, we’ve made it our mission to secure it all.

WHAT BRINGS US TOGETHER

We believe that everyone – from home computer users through to large corporations and governments – should be able to protect what matters to them most. Whether it’s privacy, family, finances, customers, business success or critical infrastructure, we’ve made it our mission to secure it all.

THREAT LANDSCAPE

KASPERSKY lab

FACTS ABOUT US

Founded in 1997 and led by Eugene Kaspersky

Holding company registered in the United Kingdom

Provides innovative IT security solutions for business and consumers

>20 million products Activation per year

711 million USD — global Unaudited revenue in 2014

>3,000 highly qualified specialists

One of the four biggest endpoint security vendors*

“Leader” according to the Gartner Magic Quadrant for Endpoint Protection Platforms**

Our solutions are recognized and awarded in independent tests and reviews

GEOGRAPHY

200 countries and territories in which we operate

34 representative regional offices

North America

Canada
Mexico
USA

South America

Brazil

Africa

South Africa

Europe

Austria
Denmark
France
Germany
Israel
Italy
Netherlands
Poland
Portugal
Romania
Russia (HQ)
Spain
Switzerland
UK
Ukraine

Asia

China
India
Hong Kong
Japan
Kazakhstan
Malaysia
South Korea
Singapore
Turkey
UAE

Australia

THREAT LANDSCAPE

KASPERSKY lab

OUR TRANSPARENCY PRINCIPLES

We host the annual Kaspersky Lab Security Analyst Summit which brings together the world's best IT security experts

We hold regular training courses for INTERPOL and Europol officers and the police forces of many countries, e.g. City of London Police

We provide expert speakers at conferences around the globe, e.g. World Economic Forum in Davos

OUR TRANSPARENCY PRINCIPLES

Data sent to Kaspersky Lab is **depersonalized and does not include** users' confidential information

We detect and neutralize **threats, regardless of their origin or purpose**

We work with **governments and law enforcement agencies** to fight cyberthreats

We are committed to **the trustworthy development** of our technologies and solutions

We cooperate with the **IT securiitty industry in joint cyberthreat investigations**

SCHEMES
CHARTS
AND DIAGRAMS

KASPERSKY lab

ORG CHART

Eugene Kaspersky

Chief Executive Officer

Andrey Tikhonov

Chief Operative Officer

Nikita Shvetsov

Chief Technology Officer

Alexey De-Monderik

Chief Investment Officer, Corporate Advisor

Denis Zenkin

Head of Corporate Communications

Igor Cherkunov

Chief Legal Officer

Andrey Tikhonov

Chief Operative Officer

Nikita Shvetsov

Chief Technology Officer

Alexey De-Monderik

Chief Investment Officer, Corporate Advisor

Denis Zenkin

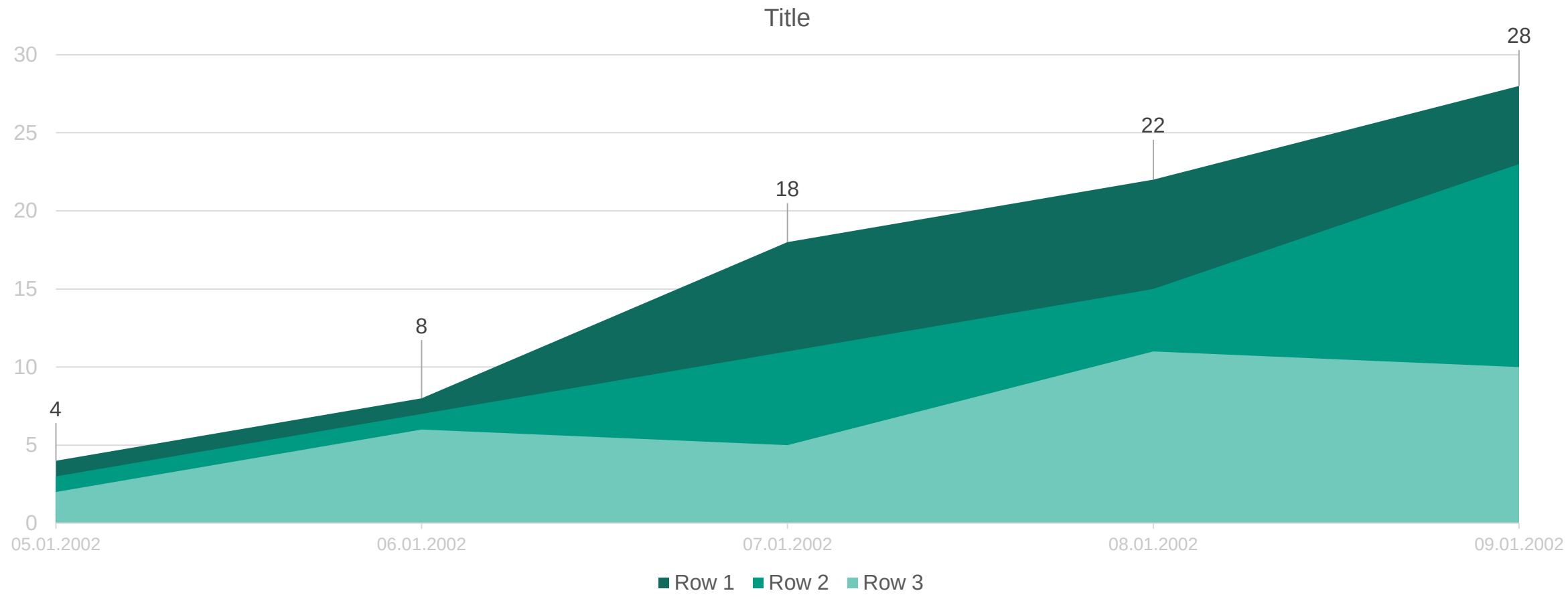
Head of Corporate Communications

Igor Cherkunov

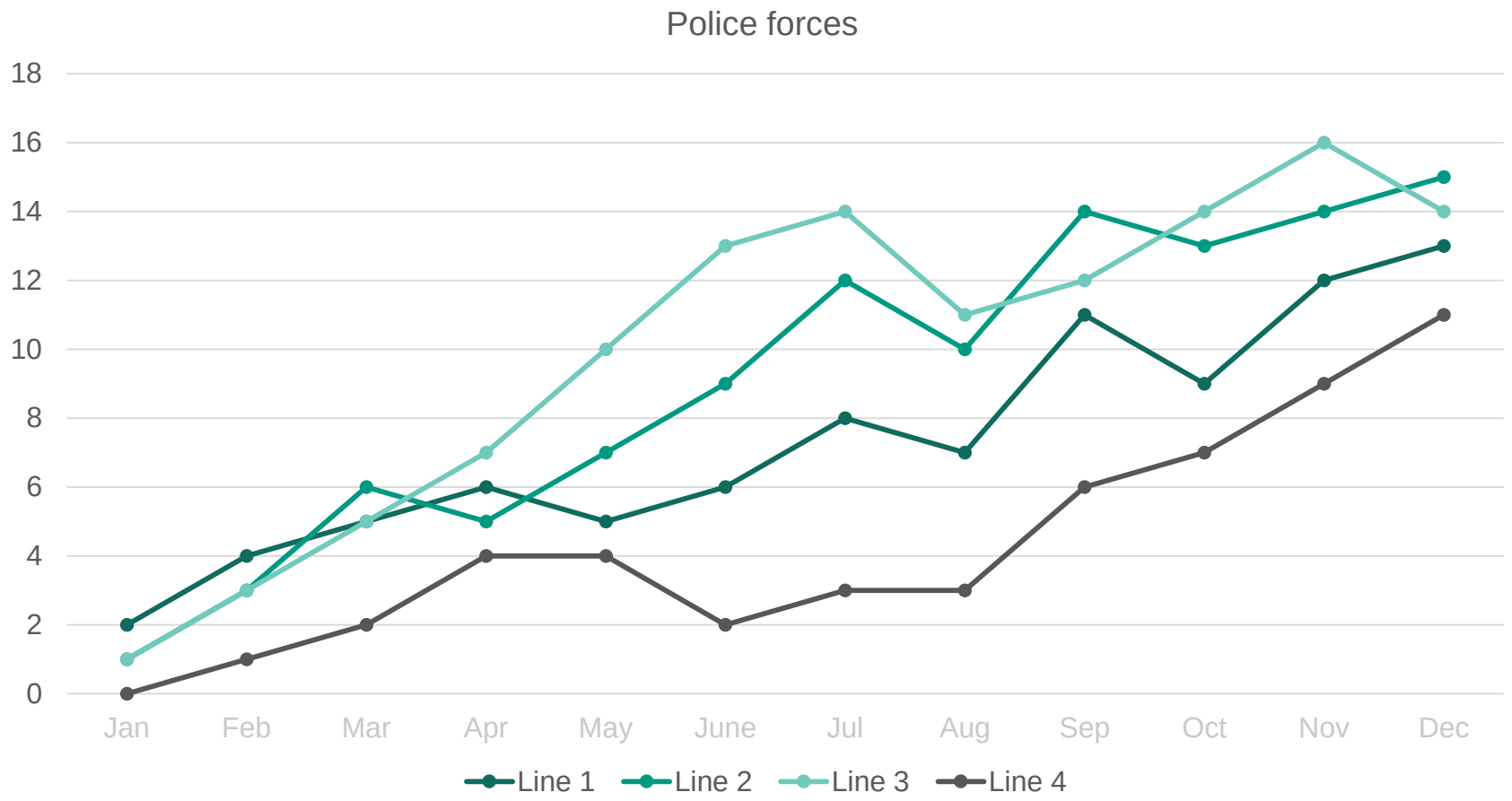
Chief Legal Officer

SCHEMES & CHARTS 1

We hold regular training courses for INTERPOL and Europol officers and the police forces of many countries, e.g. City of London Police



SCHEMES & CHARTS 2



8.243

Sales in 12 month

Ut wisi enim lorem station nostrud exenrtiz

7.306

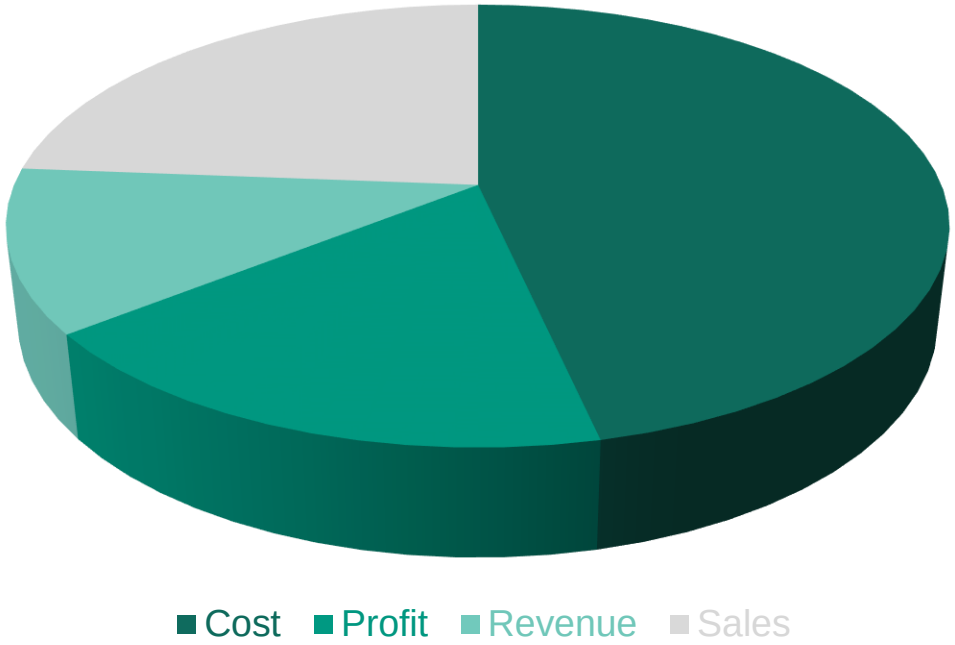
Rafferals in 12 month

Ut wisi enim lorem station nostrud exenrtiz

We hold regular training courses for INTERPOL and Europol officers and the police forces of many countries, e.g. City of London Police

SCHEMES & CHARTS 3

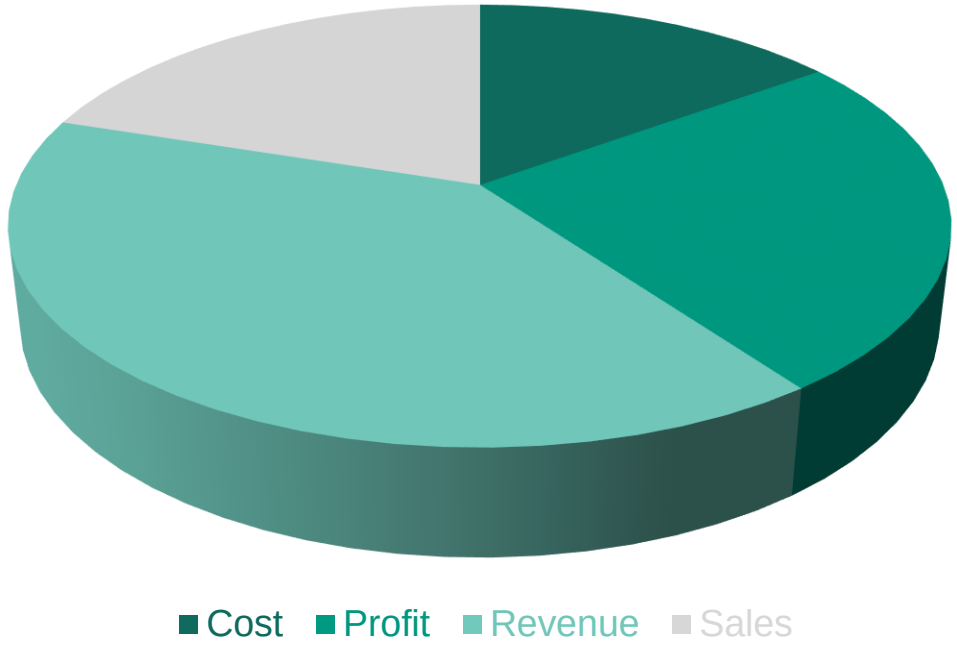
Chart 1



Controlled

Mauris quam dolor, cursus at porta et, luctus eget purus. Nunc tempor luctus interdum. Duis libero leo, consequat ut accumsan eu, viverra et, consequat ut accumsan erat.

Chart 2



Controlled

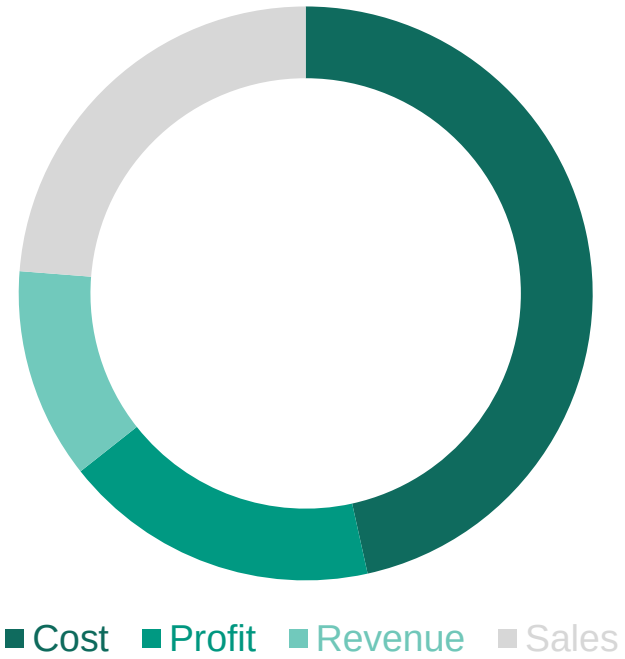
Mauris quam dolor, cursus at porta et, luctus eget purus. Nunc tempor luctus interdum. Duis libero leo, consequat ut accumsan eu, viverra et, consequat ut accumsan erat.

SCHEMES & CHARTS 4

Controlled

Mauris quam dolor, cursus at porta et, luctus eget purus. Nunc tempor luctus interdum. Duis libero leo, consequat ut accumsan eu, viverra et, consequat ut accumsan erat.

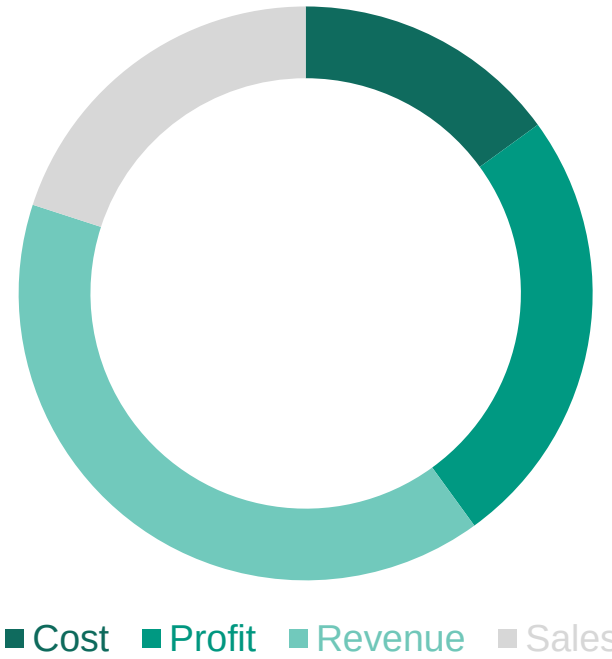
Chart 1



Controlled

Mauris quam dolor, cursus at porta et, luctus eget purus. Nunc tempor luctus interdum. Duis libero leo, consequat ut accumsan eu, viverra et, consequat ut accumsan erat.

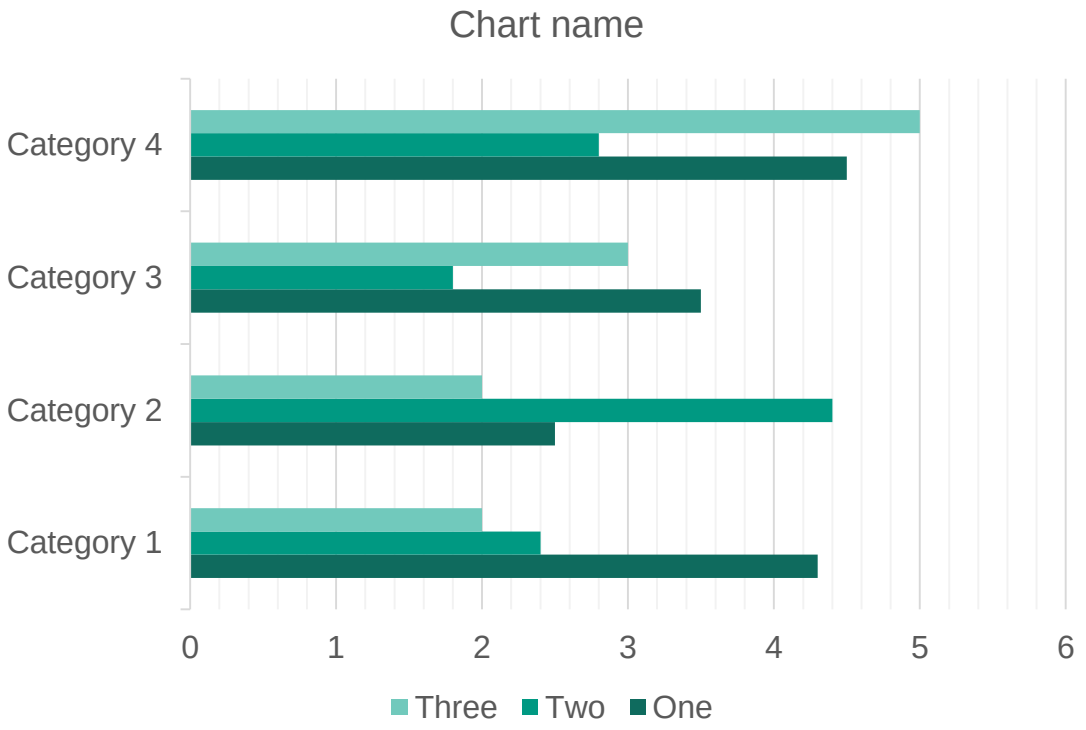
Chart 2



SCHEMES & CHARTS 5

Controlled

Mauris quam dolor, cursus at porta et, luctus eget purus. Nunc tempor luctus interdum. Duis libero leo, consequat ut accumsan eu, viverra et, consequat ut accumsan erat.



Controlled

Mauris quam dolor, cursus at porta et, luctus eget purus. Nunc tempor luctus interdum. Duis libero leo, consequat ut accumsan eu, viverra et, consequat ut accumsan erat.

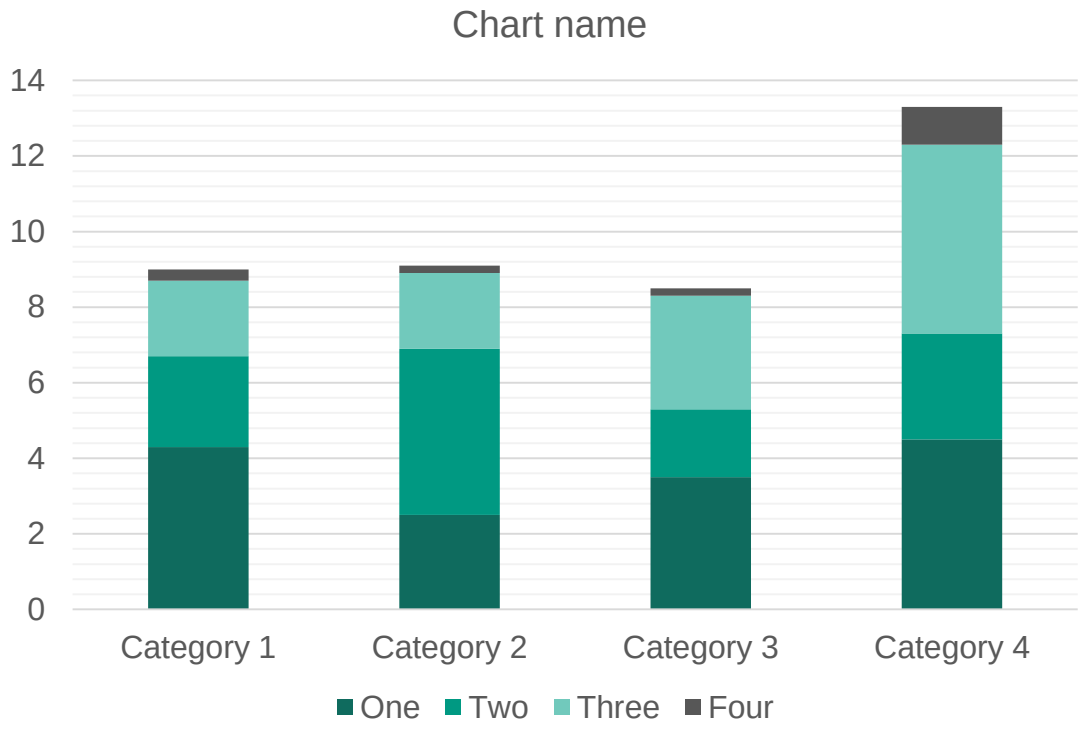


TABLE 1

We hold regular training courses for INTERPOL and Europol officers and the police forces of many countries, e.g. City of London Police

NAME	January	February	March	April	May
JOHN	$2,000	$1,800	$1,900	$3,100	$3,000
ALEX	$2,300	$2,000	$2,400	$1,900	$3,000
JENNY	$5,000	$1,900	$5,000	$2,400	$2,400
GEORGE	$2,300	$500	$1,000	$350	$800
SUNNY	$2,000	$2,400	$0	$2,400	$3,000
ALEX	$2,300	$2,000	$1,000	$1,900	$3,000
MAX	$5,000	$1,900	$2,200	$1,000	$2,400

LET’S TALK?

Kaspersky Lab HQ
39A/3 Leningradskoe Shosse
Moscow, 125212, Russian Federation
Tel: +7 (495) 797-8700
www.kaspersky.com

KASPERSKY lab