Object Oriented Code RE with HexRaysCodeXplorer

Eugene Rodionov
@vxradius

Alex Matrosov
@matrosov

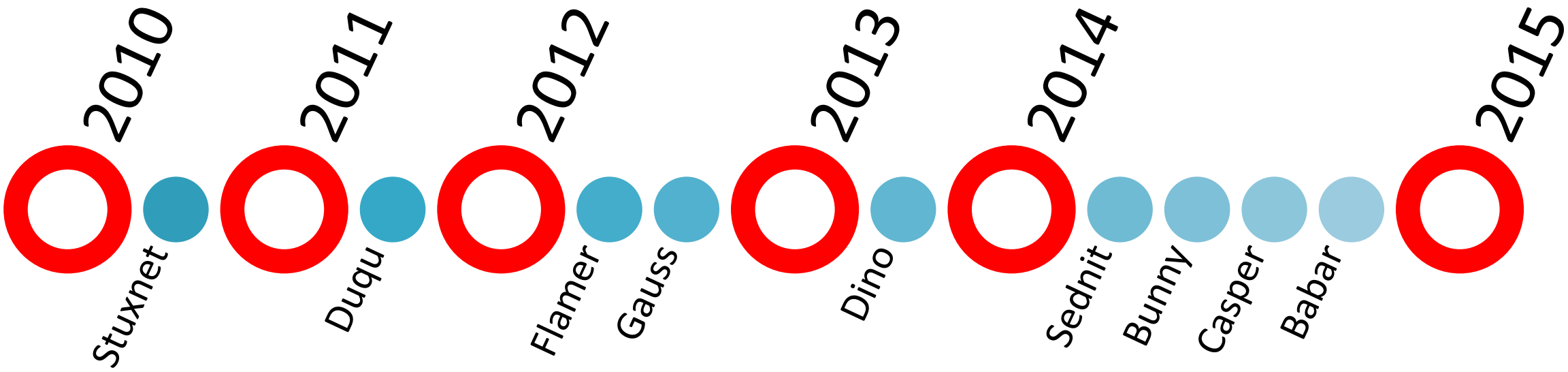
Agenda

* Object Oriented Code Reversing Challenges
  -- virtual methods
  -- templates

* Reversing Object Oriented Malware
  -- Flamer
  -- Sednit

* HexRaysCodeXplorer in use

Modern C++ Malware for Targeted Attacks



nsec

Why reversing C++ code is a hard problem?

Virtual Methods & Templates

nsec

Virtual Methods

```
class Cat {
private:
   int _weight;
public:
   Cat(int weight) : _weight(weight) {};

   int eat(int food) {
      return _weight += food;
   };
};

int _tmain(int argc, _TCHAR* argv[])
{
   Cat* cat = new Cat(130);
   int newWeigth = cat->eat(20);
}
```

vs

```
class Animal {
protected:
   int _weight;
public:
   Animal(int weight) : _weight(weight) {};
   virtual int eat(int food) = 0;
};

class Cat : Animal {
public:
   Cat(int weight) : Animal(weight) {};

   virtual int eat(int food) {
      return _weight += food;
   };
};

int _tmain(int argc, _TCHAR* argv[])
{
   Animal* cat = new Cat(130);
   int newWeight = cat->eat(20);
}
```

Virtual Methods

```
int __cdecl wmain()
{
  memset(&v2, 0xCCu, 0xF4u);
  v4 = (Cat *)operator new(4u);  // allocate object Cat
  v6 = 0;
  if ( v4 )
  {
    Cat::Cat(v4, 130);           // initialize object Cat
    v2 = v0;
  }
  else
  {
    v2 = 0;
  }
  v3 = v2;
  v6 = -1;
  cat = v2;
  Cat::eat(v2, 20);              // call method eat
  return 0;
}
```

vs

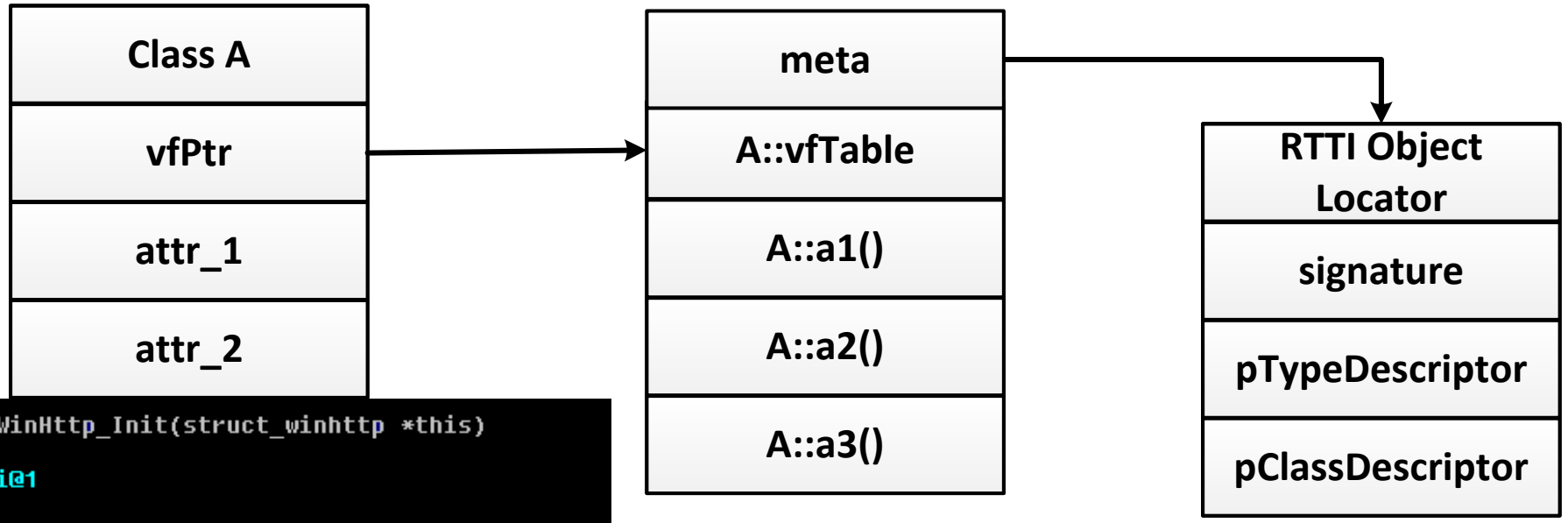
```
int __cdecl wmain()
{
  v5 = (Cat *)operator new(8u);  // allocate object Cat
  v7 = 0;
  if ( v5 )
  {
    Cat::Cat(v5, 130);           // initialize object Cat
    v3 = v0;
  }
  else
  {
    v3 = 0;
  }
  v4 = v3;
  v7 = -1;
  cat = v3;
  v3->vfptr->eat(v3, 20);        // call eat method
  return 0;
}
```

```
virtual int eat(int food) {
```

```
void __thiscall Cat::Cat(Cat *this, int weight)
{
  char v2; // [sp+Ch] [bp-CCh]@1
  Cat *const thisa; // [sp+D0h] [bp-8h]@1

  memset(&v2, 0xCCu, 0xCCu);
  thisa = this;
  Animal::Animal((Animal *)&this->vfptr, weight);
  thisa->vfptr = (AnimalVtbl *)&Cat::`vftable';
}
```

```
}
```

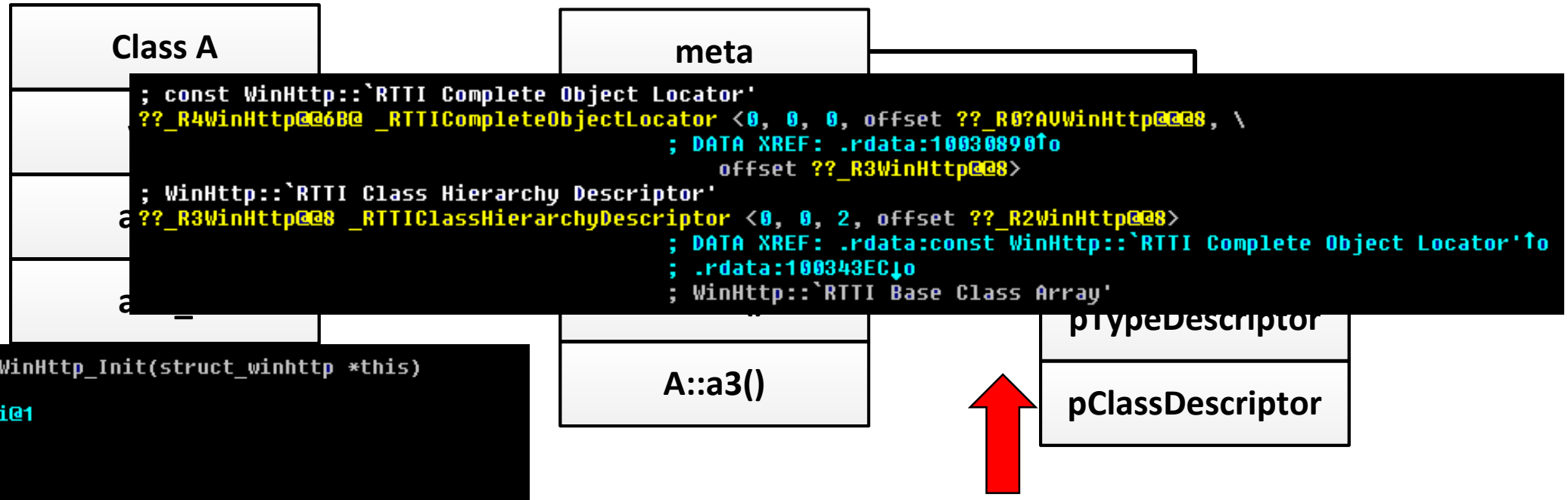
Virtual Function Tables




```
struct_winhttp *__thiscall WinHttp_Init(struct_winhttp *this)
{
  struct_winhttp *v1; // esi@1

  v1 = this;
  this->field_1 = 0;
  this->field_2 = 0;
  this->field_3 = 0;
  this->field_4 = 0;
  this->field_5 = 0;
  this->vftbl_0 = (struct_winhttp_VTABLE_0 *)&WinHttp::`vftable';
  this->field_10 = 7;
  this->field_9 = 0;
  this->field_8 = 0;
  this->field_12 = 0;
  this->field_11 = 0;
  this->field_7 = 15;
  this->field_6 = (int)calloc(0xFu, 1u);
  sub_100131F0(v1);
  sub_10011520();
  return v1;
}
```

```
.rdata:10030890 ; class WinHttp: IAgentChannel;
.rdata:10030890                 dd offset ??_R4WinHttp@@6B@
.rdata:10030894 ; const WinHttp::`vftable'
.rdata:10030894 ??_7WinHttp@@6B@ dd offset sub_100128B0
.rdata:10030898                 dd offset sub_10012DB0
.rdata:1003089C                 dd offset sub_10011320
.rdata:100308A0                 dd offset sub_100113E0
.rdata:100308A4                 dd offset sub_100113F0
.rdata:100308A8                 dd offset sub_10012D70
.rdata:100308AC                 dd offset sub_100130B0
```

Virtual Function Tables




```
struct_winhttp *__thiscall WinHttp_Init(struct_winhttp *this)
{
  struct_winhttp *v1; // esi@1

  v1 = this;
  this->field_1 = 0;
  this->field_2 = 0;
  this->field_3 = 0;
  this->field_4 = 0;
  this->field_5 = 0;
  this->vftbl_0 = (struct_winhttp_VTABLE_0 *)&WinHttp::`vftable';
  this->field_10 = 7;
  this->field_9 = 0;
  this->field_8 = 0;
  this->field_12 = 0;
  this->field_11 = 0;
  this->field_7 = 15;
  this->field_6 = (int)calloc(0xFu, 1u);
  sub_100131F0(v1);
  sub_10011520();
  return v1;
}
```

```
.rdata:10030890 ; class WinHttp: IAgentChannel;
.rdata:10030890                 dd offset ??_R4WinHttp@@6B@
.rdata:10030894 ; const WinHttp::`vftable'
.rdata:10030894 ??_7WinHttp@@6B@ dd offset sub_100128B0
.rdata:10030898                 dd offset sub_10012DB0
.rdata:1003089C                 dd offset sub_10011320
.rdata:100308A0                 dd offset sub_100113E0
.rdata:100308A4                 dd offset sub_100113F0
.rdata:100308A8                 dd offset sub_10012D70
.rdata:100308AC                 dd offset sub_100130B0
```

Virtual Function Tables

* lead to indirect method calls
  -- difficult to analyze statically

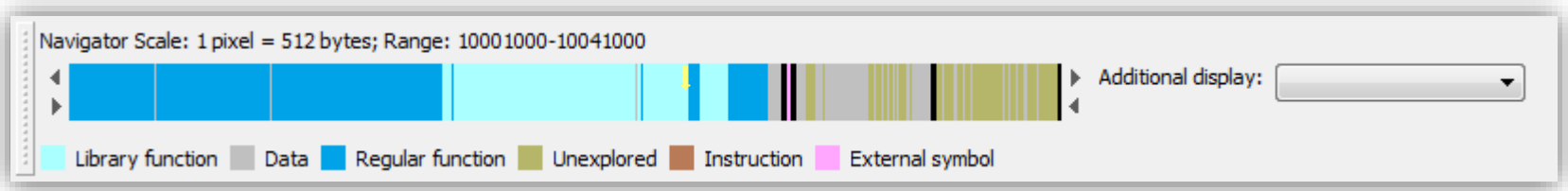
* initialized in constructors
  -- need to track back object creation

C++ Templates

* extra code to analyze
  -- another way to create polymorphic types

```
std::vector<int>                std::vector<char>
std::vector<std::string>        std::vector<custom_type>
```

* problematic to recognize standard library code (FLIRT)
  -- playing with compiler optimization options



nsec

C++ Code Reconstruction Problems

* Object identification
  -- type reconstruction

* Class layout reconstruction
  -- Identify constructors/destructors
  -- Identify class members
  -- Local/global type reconstruction
  -- Associate object with exact method calls

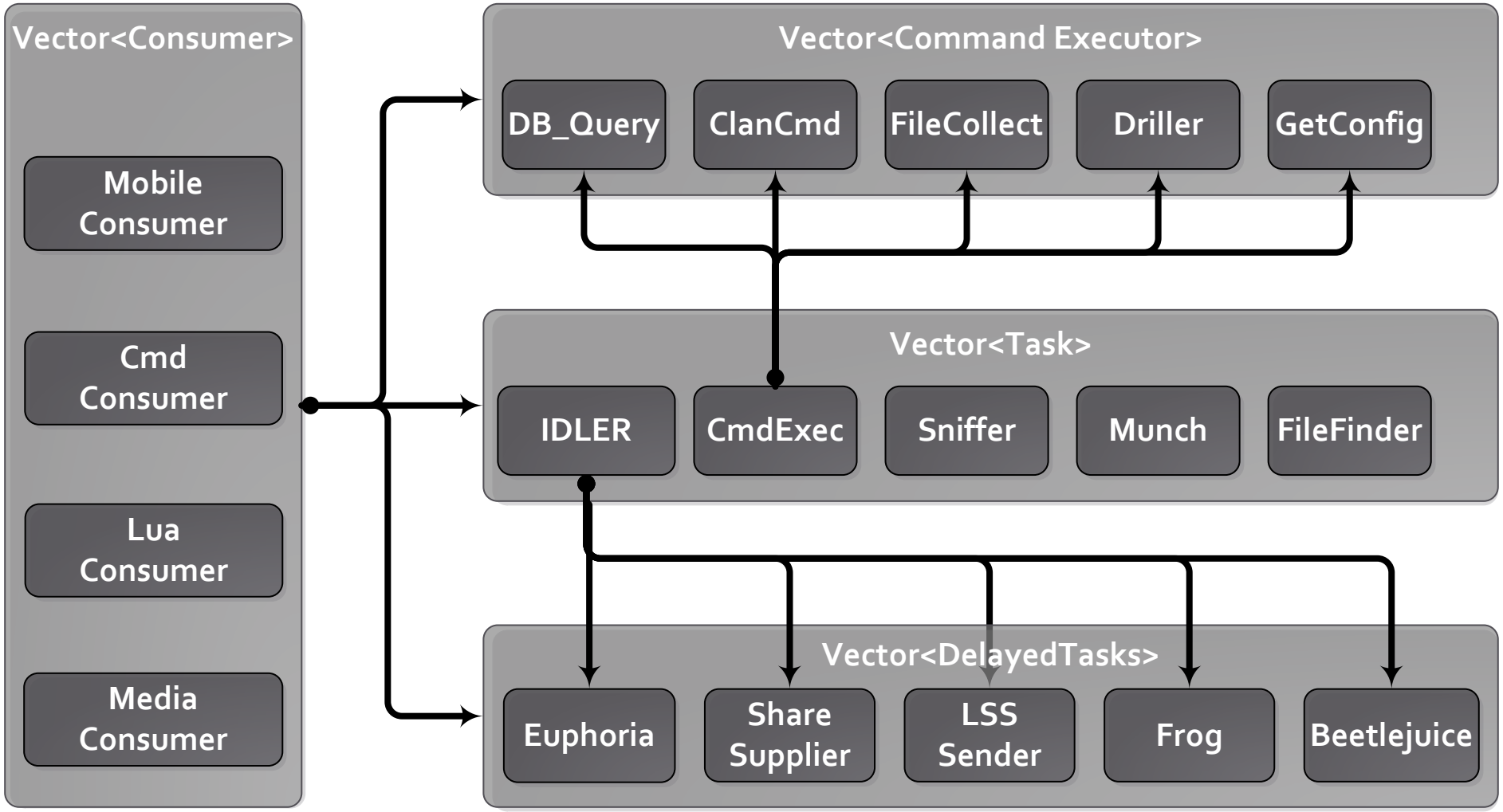
* RTTI reconstruction
  -- vftable reconstruction
  -- Associate vftable object with exact object
  -- class hierarchy reconstruction

Reversing Object Oriented Malware

Practical Approaches: REconstructing Flamer Framework

nsec

REconstructing Flamer Framework



http://www.welivesecurity.com/2012/08/02/flamer-analysis-framework-reconstruction/

nsec

REc


```
 0 0x10256aa0 - 0x10256afc:  VECTOR_DATA_2_VTABLE   method count: 23
 1 0x10256bb0 - 0x10256bd8:  FILE_MAPPING_1_VTABLE   method count: 10
 2 0x10256bd8 - 0x10256bf0:  GLOBAL_EVENT_1_VTABLE   method count: 6
 3 0x102679a0 - 0x102679f0:  PROCESS_HANDLE_VTABLE   method count: 20
 4 0x10267a90 - 0x10267acc:  THREAD_HANDLE_VTABLE   method count: 15
 5 0x10267b08 - 0x10267b7c:  FILE_VTABLE_0   method count: 29
 6 0x10267bc0 - 0x10267bd8:  EVENT_VTABLE   method count: 6
 7 0x10267df0 - 0x10267e40:  PROCESS_HANDLE_VTABLE_0   method count: 20
 8 0x10267e40 - 0x10267e80:  EVENTGLOBAL_HZ_VTABLE   method count: 16
 9 0x10267e90 - 0x10267eb0:  KASPER_EVENT_ENTRY_VTABLE   method count: 8
10 0x10267f10 - 0x10267f34:  TOKEN_HANDLE_VTABLE   method count: 9
11 0x10268118 - 0x10268120:  USTRING_REG_PATH_VTABLE   method count: 2
12 0x10268128 - 0x102681a4:  FILE_1_vTable   method count: 31
13 0x10268260 - 0x10268298:  ENC_2_VTABLE   method count: 14
14 0x10268478 - 0x102684a8:  ZLIB_HLPR_VTABLE   method count: 12
15 0x102684e0 - 0x1026853c:  ENC_3_VTABLE   method count: 23
16 0x1026856c - 0x10268590:  SYSTEM_HANDLE_INFO_VTABLE   method count: 9
17 0x10268688 - 0x102686bc:  DICT_1_VTABLE   method count: 13
18 0x10268d78 - 0x10268dd4:  MAIN_VECT_3_VTABLE   method count: 23
19 0x10268f80 - 0x10268fe8:  CONCOL_HANDLER_VTABLE   method count: 26
20 0x102693c0 - 0x102693d0:  CMD_EXECUTER_VIPER_VTABLE   method count: 4
21 0x10269490 - 0x102694ec:  MAIN_VECT_1_VTABLE   method count: 23
22 0x102694f0 - 0x1026954c:  MAIN_VECT_2_VTABLE   method count: 23
23 0x10269550 - 0x102695ac:  MAIN_VECT_4_VTABLE   method count: 23
24 0x10269768 - 0x102697dc:  MAIN_VECT_2_IDLER_VTABLE   method count: 29
25 0x102697dc - 0x10269818:  _MAIN_VECT_2_IDLER_VTABLE   method count: 15
26 0x10269818 - 0x10269874:  VECT_VTABLE   method count: 23
27 0x10269874 - 0x10269884:  MAIN_VECT_4_TIME_UPDATER_VTABLE   method count: 4
28 0x10269a2c - 0x10269a68:  MAIN_3_VECT_1_VTABLE   method count: 15
29 0x10269b48 - 0x10269bbc:  MAIN_VECT_2_HNT_VTABLE   method count: 29
30 0x10269bc8 - 0x10269c3c:  MAIN_VECT_2_VOLUME_SUPPLIER_VTABLE   method count: 29
31 0x10269c40 - 0x10269cb4:  MAIN_VECT_2_VIRTUAL_VOLUME_SUPPLIER_VTABLE   method count: 29
32 0x10269e10 - 0x10269e84:  MAIN_VECT_2_HeadacheConsumer_VTABLE   method count: 29
```

http://www.welivesecurity.com/2012/08/02/flamer-analysis-framework-reconstruction/

nsec

Identifying Used Types

* Smart pointers

* Strings

* Vectors to maintain objects

* Custom data types:
  -- tasks
  -- triggers
  -- and etc.

nsec

Data Types Being Used: Smart pointers

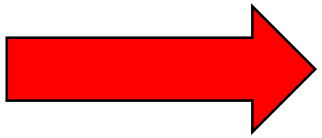
```
struct SMART_PTR
{
   void *pObject;      // pointer to the object
   int   *RefNo;       // reference counter
};
```

```
SMART_PTR_STRUCT *__userpurge SmartPtr_InializeByObject<eax>(SMART_PTR_STRUCT *a1<esi>, void *pObject)
{
  int *v2; // eax@1

  LOBYTE(v2) = new(4);
  if ( v2 )
    *v2 = 1;
  else
    v2 = 0;
  a1->RefNo = v2;
  a1->Object = pObject;
  return a1;
}
```

nsec

Data Types Being Used: Smart pointers

```
SmartPtr_InializeByObject proc near     ; CODE XR
                                        ; sub_100

var_10          = dword ptr -10h
var_C           = dword ptr -0Ch
var_4           = dword ptr -4
arg_0           = dword ptr  8

                mov     eax, offset sub_101C690A
                call    __EH_prolog
                push    ecx
                push    4
                call    alloc_mem
                pop     ecx
                mov     [ebp+var_10], eax
                and     [ebp+var_4], 0
                test    eax, eax
                jz      short loc_100041F5
                mov     dword ptr [eax], 1
                jmp     short loc_100041F7
; ---------------------------------------------------------------------------

loc_100041F5:                           ; CODE XR
                xor     eax, eax

loc_100041F7:                           ; CODE XR
                or      [ebp+var_4], 0FFFFFFFFh
                mov     ecx, [ebp+var_C]
                mov     [esi+4], eax
                mov     eax, [ebp+arg_0]
                mov     [esi], eax
                mov     eax, esi
                mov     large fs:0, ecx
                leave
                retn    4
SmartPtr_InializeByObject endp
```



```
SMART_PTR_STRUCT *__userpurge SmartPtr
{
  int *v2; // eax@1

  v2 = alloc_mem(4);
  if ( v2 )
    *v2 = 1;
  else
    v2 = 0;
  a1->RefNo = v2;
  a1->Object = a2;
  return a1;
}
```

Data Types Being Used: Vectors

```
struct VECTOR
{
  void *vTable;              // pointer to the virtual table
  int NumberOfItems;         // self-explanatory
  int MaxSize;               // self-explanatory
  void *vector;              // pointer to buffer with elements
};
```

* Used for handling objects:
    -- tasks
    -- triggers

Data Types Being Used: Strings

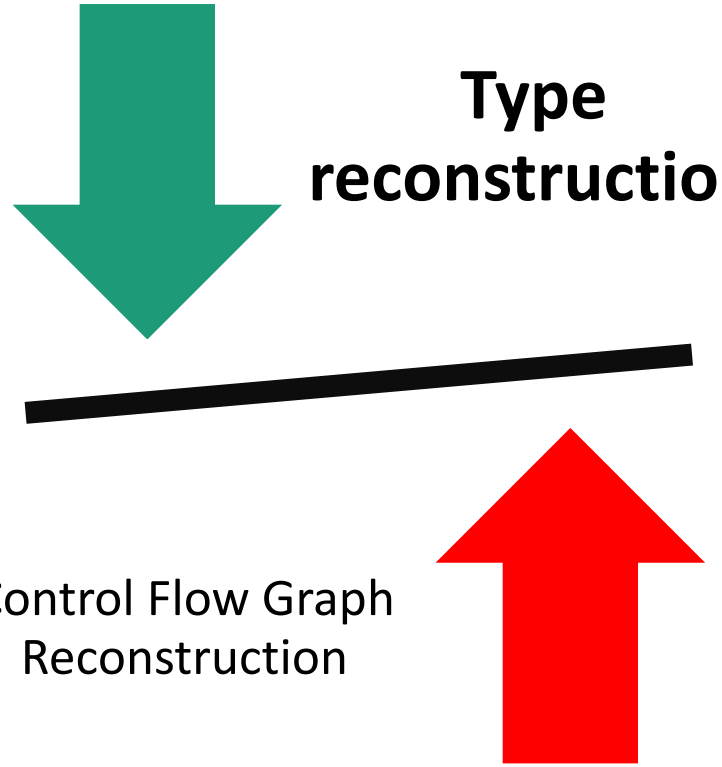
```
struct USTRING_STRUCT
{
  void *vTable;              // pointer to the table
  int RefNo;                 // reference counter
  int Initialized;
  wchar_t *UnicodeBuffer; // pointer to unicode string
  char *AsciiBuffer;         // pointer to ASCII string
  int AsciiLength;           // length of the ASCII string
  int Reserved;
  int Length;                // Length of unicode string
  int LengthMax;             // Size of UnicodeBuffer
};
```

Approaching Flamer

* Identify Object Constructors

* Reconstruct Object Attributes

* Reconstruct Object Methods



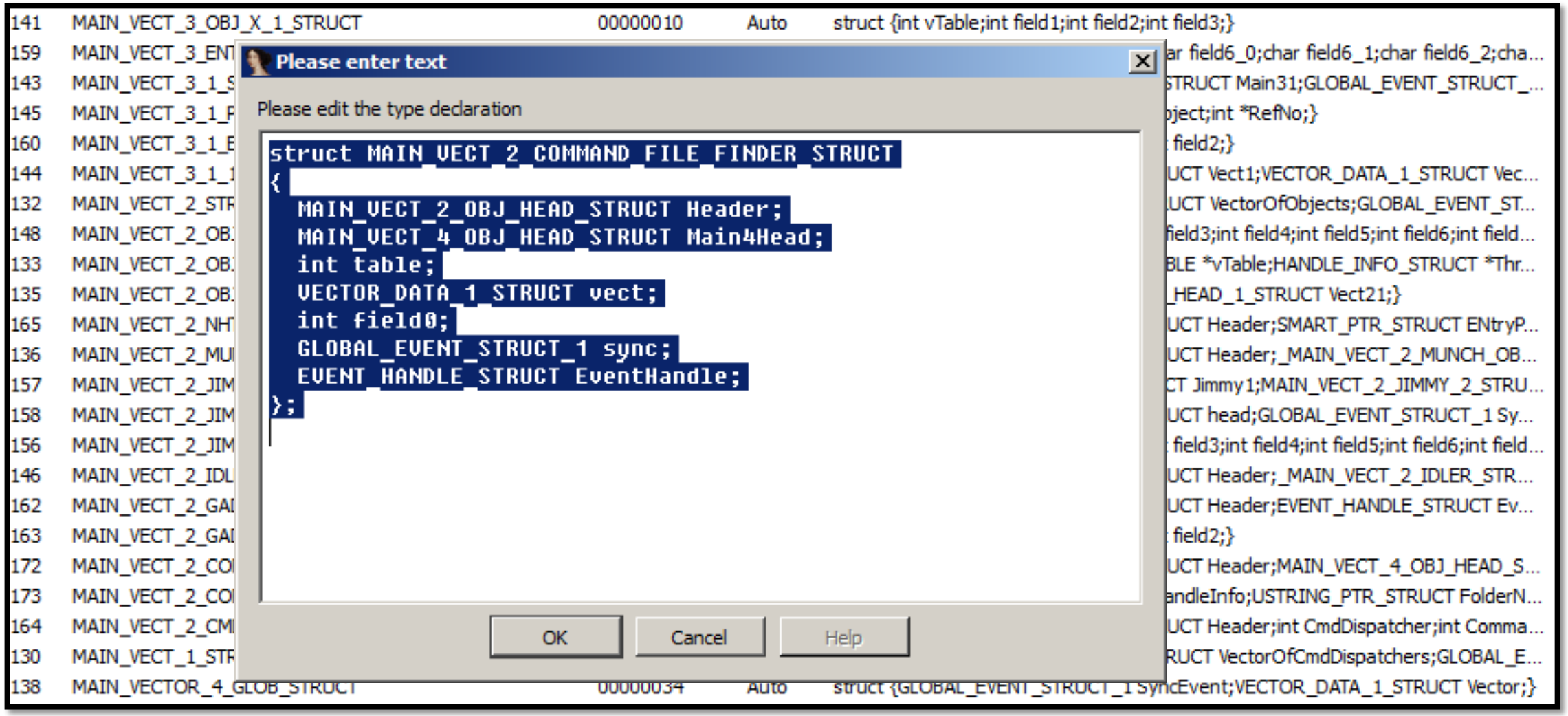
Identifying Object Constructors

```
USTRING_PTR_STRUCT *__thiscall UStringPtr_Construct(USTRING_PTR_STRUCT *this, wchar_t *String)
{
  USTRING_PTR_STRUCT *v2; // ebx@1
  USTRING_STRUCT *v3; // eax@1
  USTRING_STRUCT *v4; // eax@2

  v2 = this;
  this->vTable = UStringPtr_Vtable;
  v3 = alloc_mem(36);
  if ( v3 )
    v4 = UString_InitByWcharStr(v3, String);
  else
    v4 = 0;
  v2->String = v4;
  UStringPtr_Reinit(&v2->String, 0);
  return v2;
}
```

REconstructing Object's Attributes

141	MAIN_VECT_3_OBJ_X_1_STRUCT	00000010	Auto	struct {int vTable;int field1;int field2;int field3;}
159	MAIN_VECT_3_ENTRY	00000044		struct {int vTable;int field4;int field5;char field6_0;char field6_1;char field6_2;cha...
143	MAIN_VECT_3_1_STRUCT	0000004C	Auto	struct {int vTable;MAIN_VECT_3_1_1_STRUCT Main31;GLOBAL_EVENT_STRUCT_...
145	MAIN_VECT_3_1_PTR_STRUCT	00000008	Auto	struct {MAIN_VECT_3_1_STRUCT *pObject;int *RefNo;}
160	MAIN_VECT_3_1_ENTRY	00000010		struct {int vTable;int field0;int field1;int field2;}
144	MAIN_VECT_3_1_1_STRUCT	00000024	Auto	struct {int vTale;VECTOR_DATA_1_STRUCT Vect1;VECTOR_DATA_1_STRUCT Vec...
132	MAIN_VECT_2_STRUCT	00000080	Auto	struct {int field0;VECTOR_DATA_1_STRUCT VectorOfObjects;GLOBAL_EVENT_ST...
148	MAIN_VECT_2_OBJ_HEAD_VTABLE	00000074	Auto	struct {int field0;int field1;int field2;int field3;int field4;int field5;int field6;int field...
133	MAIN_VECT_2_OBJ_HEAD_STRUCT	00000088	Auto	struct {MAIN_VECT_2_OBJ_HEAD_VTABLE *vTable;HANDLE_INFO_STRUCT *Thr...
135	MAIN_VECT_2_OBJ_HEAD_1_STRUCT	00000028	Auto	struct {int vTable;_MAIN_VECT_2_OBJ_HEAD_1_STRUCT Vect21;}
165	MAIN_VECT_2_NHT_STRUCT	00000098	Auto	struct {MAIN_VECT_2_OBJ_HEAD_STRUCT Header;SMART_PTR_STRUCT ENtryP...
136	MAIN_VECT_2_MUNCH_OBJ_STRUCT	000000DC	Auto	struct {MAIN_VECT_2_OBJ_HEAD_STRUCT Header;_MAIN_VECT_2_MUNCH_OB...
157	MAIN_VECT_2_JIMMY_STRUCT	00000188	Auto	struct {MAIN_VECT_2_JIMMY_1_STRUCT Jimmy1;MAIN_VECT_2_JIMMY_2_STRU...
158	MAIN_VECT_2_JIMMY_2_STRUCT	00000150	Auto	struct {MAIN_VECT_2_OBJ_HEAD_STRUCT head;GLOBAL_EVENT_STRUCT_1 Sy...
156	MAIN_VECT_2_JIMMY_1_STRUCT	00000038	Auto	struct {int vTable;int field1;int field2;int field3;int field4;int field5;int field6;int field...
146	MAIN_VECT_2_IDLER_STRUCT	000000BC	Auto	struct {MAIN_VECT_2_OBJ_HEAD_STRUCT Header;_MAIN_VECT_2_IDLER_STR...
162	MAIN_VECT_2_GADGET_SUPP_STRUCT	000003DC	Auto	struct {MAIN_VECT_2_OBJ_HEAD_STRUCT Header;EVENT_HANDLE_STRUCT Ev...
163	MAIN_VECT_2_GADGET_SUPP_1_STRUCT	00000010	Auto	struct {int vTable;int field0;int field1;int field2;}
172	MAIN_VECT_2_COMMAND_FILE_FINDER_STRUCT	000000DC	Auto	struct {MAIN_VECT_2_OBJ_HEAD_STRUCT Header;MAIN_VECT_4_OBJ_HEAD_S...
173	MAIN_VECT_2_COMMAND_FILE_FINDER_NOTIF_ENTRY_...	00000014		struct {HANDLE_INFO_PTR_STRUCT HandleInfo;USTRING_PTR_STRUCT FolderN...
164	MAIN_VECT_2_CMD_RUNNER_STRUCT	0000009C	Auto	struct {MAIN_VECT_2_OBJ_HEAD_STRUCT Header;int CmdDispatcher;int Comma...
130	MAIN_VECT_1_STRUCT	000000E4	Auto	struct {int vTable;VECTOR_DATA_1_STRUCT VectorOfCmdDispatchers;GLOBAL_E...
138	MAIN_VECTOR_4_GLOB_STRUCT	00000034	Auto	struct {GLOBAL_EVENT_STRUCT_1 SyncEvent;VECTOR_DATA_1_STRUCT Vector;}

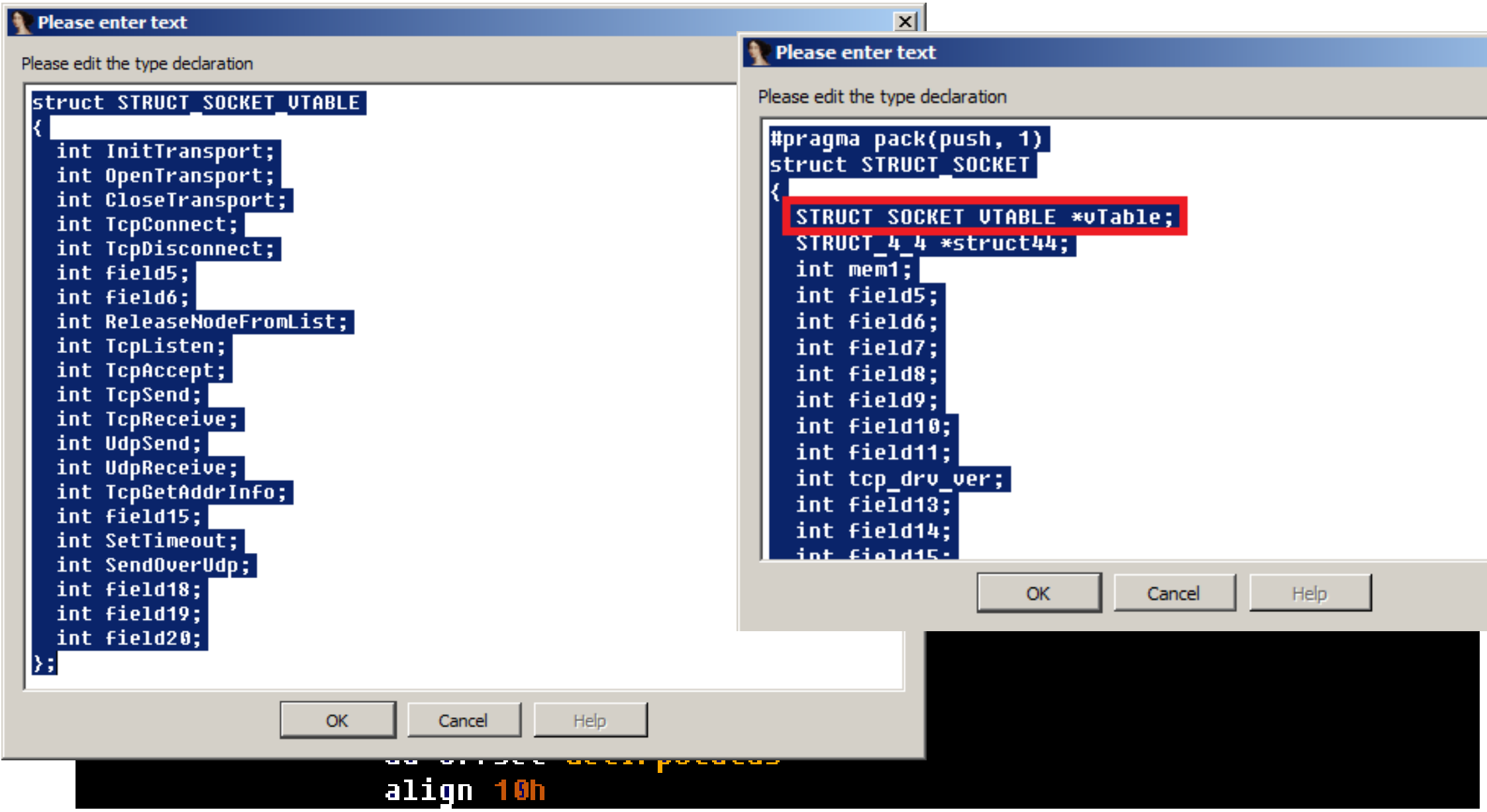
REconstructing Object's Attributes



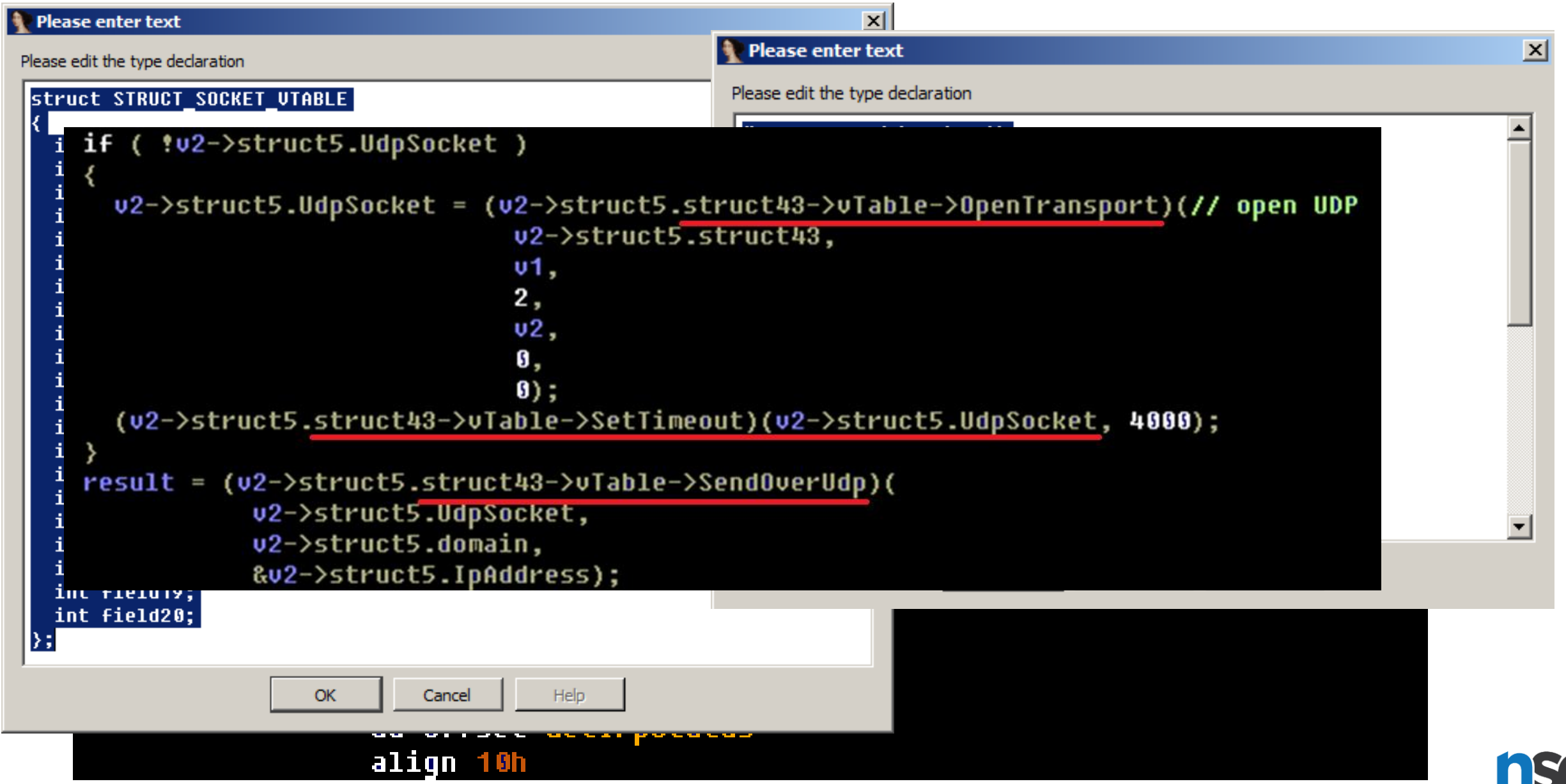
REconstructing Object's Methods

```
csocket_v_table dd offset InitializeTransport
                dd offset OpenTransport
                dd offset CloseTransport
                dd offset TcpConnect    ; returns 1 if OK and 0 - otherwise
                dd offset TcpDisconnect
                dd offset sub_1E4EF
                dd offset sub_1E510
                dd offset ReleaseNodeFromList
                dd offset TcpListen
                dd offset TcpAccept
                dd offset TcpSend
                dd offset TcpReceive
                dd offset UdpSend
                dd offset ReceiveDataFromUdp
                dd offset GetTcpAddressInfo
                dd offset sub_1E5A8
                dd offset SetTimeout
                dd offset SendOverUdp
                dd offset ret_0
                dd offset GetErrorCode
                dd offset GetIrpStatus
                align 10h
```

REconstructing Object's Methods

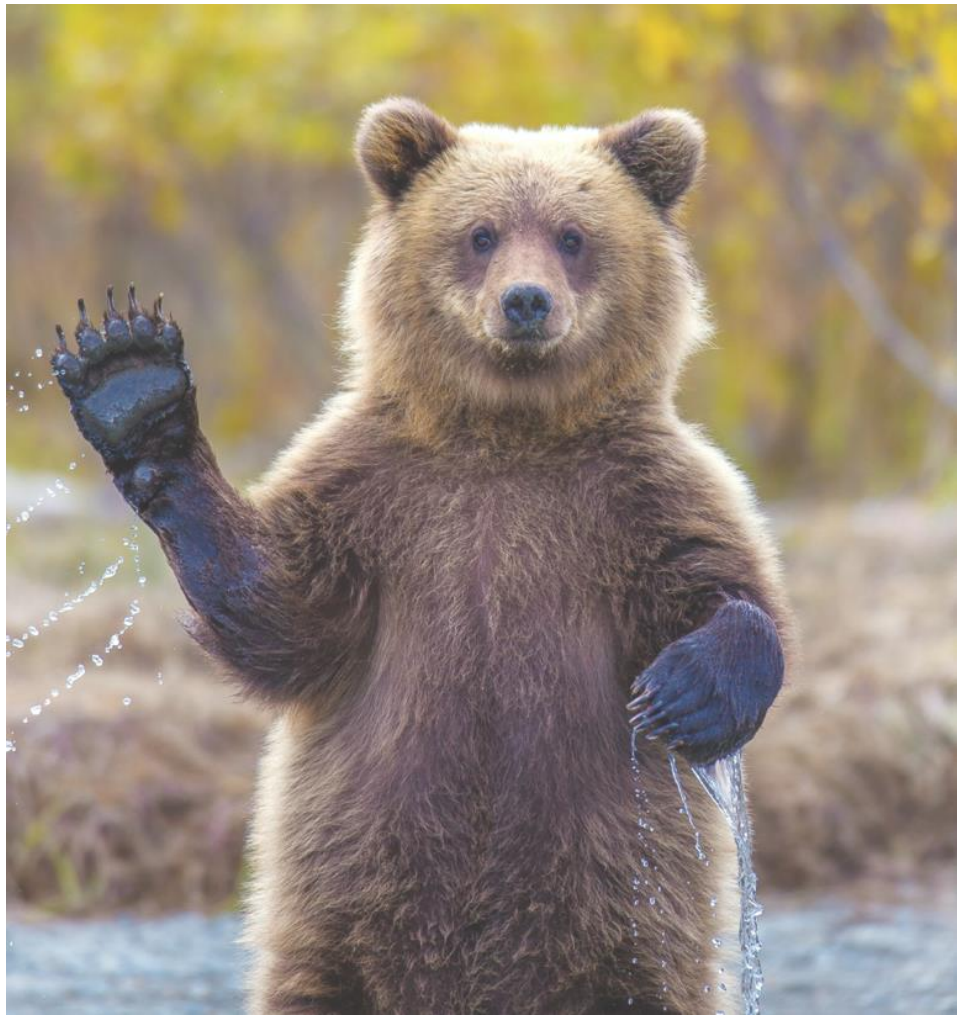


REconstructing Object's Methods



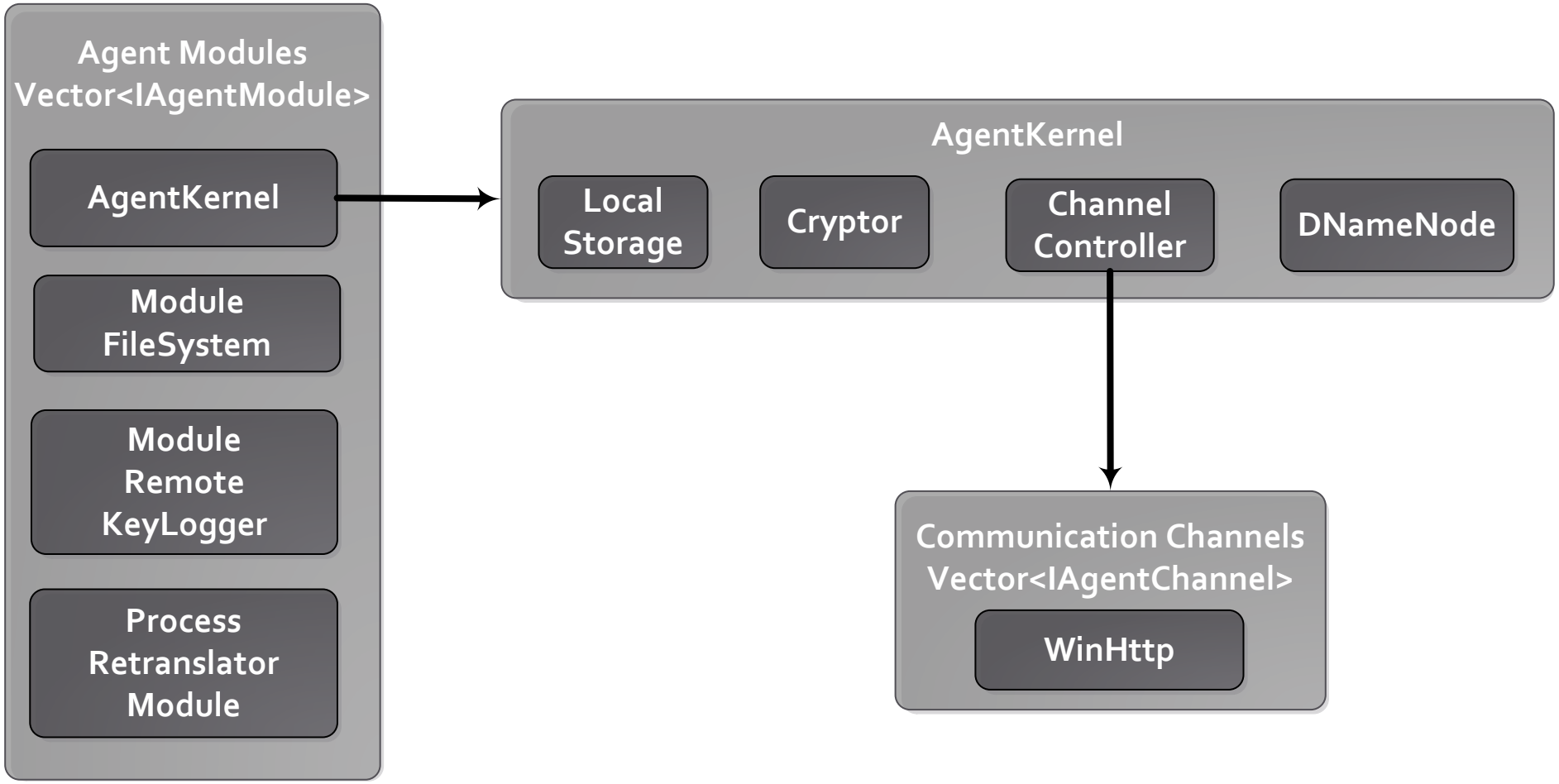
Reversing Object Oriented Malware

Practical Approaches: REconstructing XAgent Framework



nsec

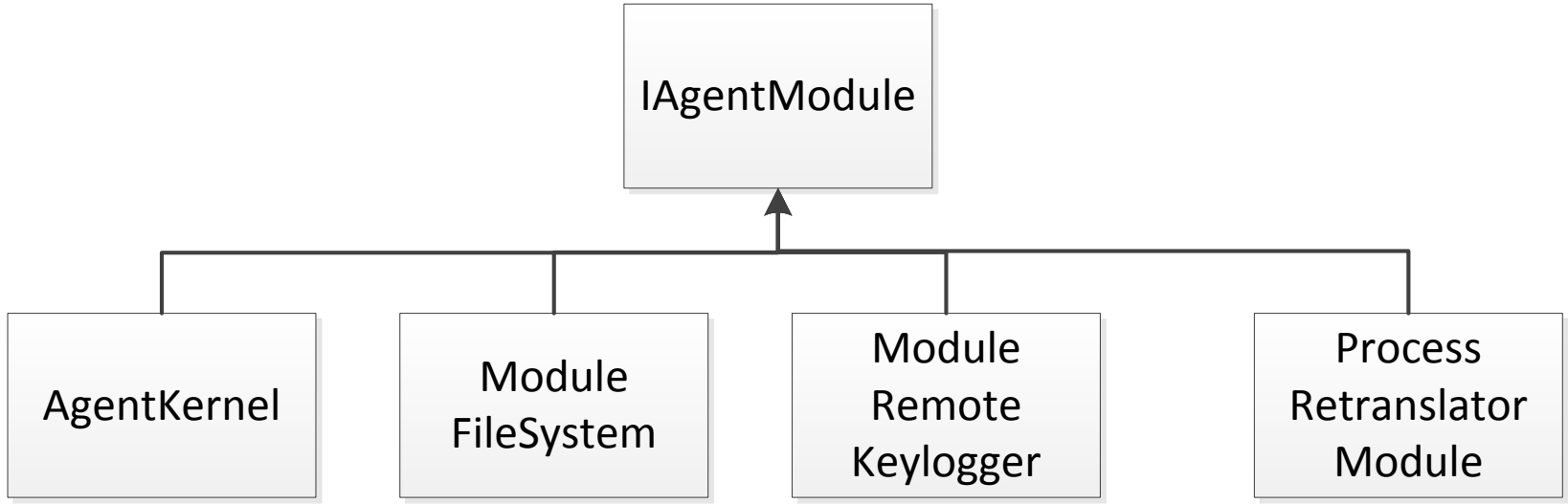
XAgent Framework



http://www.welivesecurity.com/2014/10/08/sednit-espionage-group-now-using-custom-exploit-kit/

Object Interconnection: IAgentModule

```
struct IAgentModule {
    LPVOID receiveMessage;
    LPVOID sendMessage;
    LPVOID getModuleId;
    LPVOID setModuleId;
    LPVOID executeModule;
};
```



Exploring RTTI*

Vftable	Methods	Flags	Type	Hierarchy
1003057C	5	M	AgentKernel	IKernelProvider:
10030594	5	M	AgentKernel	AgentKernel: IAgentModule, IKernelProvider;
100309D4	6		CClassFactory	CClassFactory: CUnknown<IClassFactory>, struct IClassFactory, struct IUnknown;
10030A10	7		CEventSink	CEventSink: struct DWebBrowserEvents2, struct IDispatch, struct IUnknown;
100309B8	6		CObjectWithSite	CObjectWithSite: CUnknown<IObjectWithSite>, struct IObjectWithSite, struct IUnknown;
10030980	6		CUnknown<IClassFactory>	CUnknown<IClassFactory>: struct IClassFactory, struct IUnknown;
1003099C	6		CUnknown<IObjectWithSite>	CUnknown<IObjectWithSite>: struct IObjectWithSite, struct IUnknown;
10030518	7		ChannelController	ChannelController: IChannelController;
100308FC	3		Cryptor	Cryptor: ICryptor;
100305F4	2		Gdiplus::Bitmap	Gdiplus::Bitmap: Gdiplus::Image, Gdiplus::GdiplusBase;
100305E8	2		Gdiplus::Image	Gdiplus::Image: Gdiplus::GdiplusBase;
10030538	5		IKernelProvider	IKernelProvider:
1003090C	4		ILocalDataStorage	ILocalDataStorage:
10030858	2		IPTExternChannel	IPTExternChannel:
10030920	4	M	LocalStorage	ILocalDataStorage:
10030934	5	M	LocalStorage	LocalStorage: ILocalParamStorage, ILocalDataStorage;
100305CC	5		ModuleFileSystem	ModuleFileSystem: IAgentModule;
10030840	5		ModuleRemoteKeyLogger	ModuleRemoteKeyLogger: IAgentModule;
10030864	2	M	ProcessRetranslatorModule	IPTExternChannel:
10030870	5	M	ProcessRetranslatorModule	ProcessRetranslatorModule: IAgentModule, IPTExternChannel;
10030954	3		ReservedApi	ReservedApi: IReservedApi;
10030894	7		WinHttp	WinHttp: IAgentChannel;

** IDA ClassInformer plugin*

nsec

Exploring RTTI*

Vftable	Methods	Flags	Type	Hierarchy
1003057C	5	M	AgentKernel	IKernelProvider:
10030594	5	M	AgentKernel	AgentKernel: IAgentModule, IKernelProvider;
100309D4	6		CClassFactory	CClassFactory: CUnknown<IClassFactory>, struct IClassFactory, struct IUnknown;
10030A10	7		CEventSink	CEventSink: struct DWebBrowserEvents2, struct IDispatch, struct IUnknown;

```
.rdata:1003083C ; class ModuleRemoteKeyLogger: IAgentModule;   (#classinformer)
.rdata:1003083C                 dd offset ??_R4ModuleRemoteKeyLogger@@6B@ ; const ModuleRemoteKeyLogger::`RTTI Complete Object Locator'
.rdata:10030840 ; const ModuleRemoteKeyLogger::`vftable'
.rdata:10030840 ??_7ModuleRemoteKeyLogger@@6B@ dd offset ModuleRemoteKeyLogger_recvMessage
.rdata:10030840                                         ; DATA XREF: ModuleRemoteKeyLogger+3E↑o
.rdata:10030844                 dd offset ModuleRemoteKeyLogger__sendMessage
.rdata:10030848                 dd offset IAgentModule_getId
.rdata:1003084C                 dd offset IAgentModule_setId
.rdata:10030850                 dd offset ModuleRemoteKeyLogger_run
```

10030934	5	M	LocalStorage	LocalStorage: ILocalParamStorage, ILocalDataStorage;
100305CC	5		ModuleFileSystem	ModuleFileSystem: IAgentModule;
10030840	5		ModuleRemoteKeyLogger	ModuleRemoteKeyLogger: IAgentModule;
10030864	2	M	ProcessRetranslatorModule	IPTExternChannel:
10030870	5	M	ProcessRetranslatorModule	ProcessRetranslatorModule: IAgentModule, IPTExternChannel;
10030954	3		ReservedApi	ReservedApi: IReservedApi;
10030894	7		WinHttp	WinHttp: IAgentChannel;

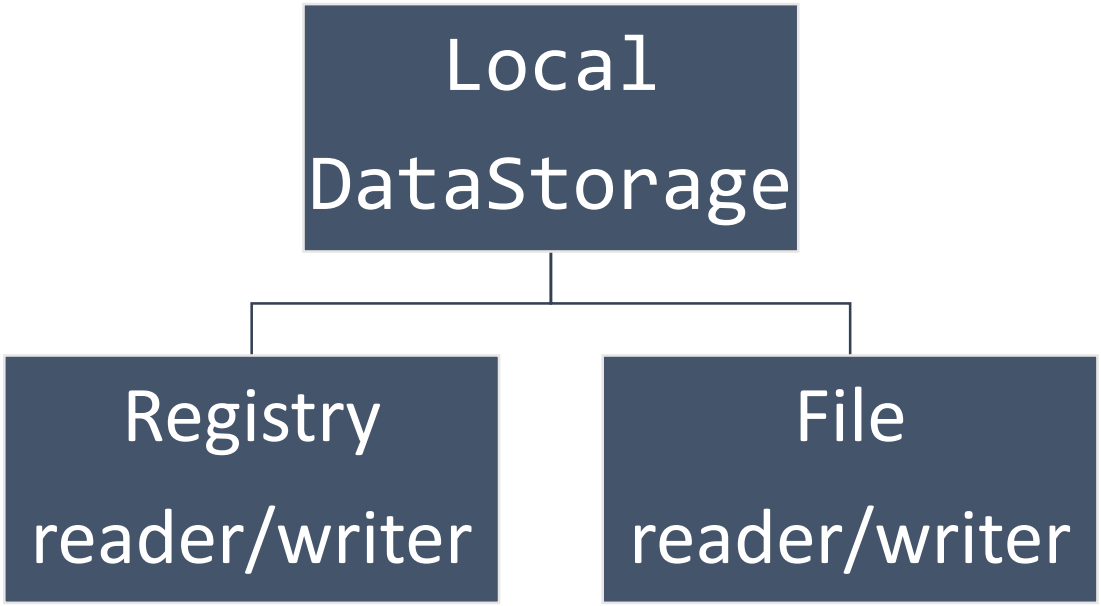
** IDA ClassInformer plugin*

nsec

XAgent: LocalDataStorage

10030920 4	M	LocalStorage	ILocalDataStorage:
10030934 5	M	LocalStorage	LocalStorage: ILocalParamStorage, ILocalDataStorage;

```
struct_local_data_storage *__thiscall LocalDataStorage_Init(struct_local_data_storage *this, void *a2, void *a3, int a4, int a5, int a6, int a7)
{
  v7 = this;
  this->vftbl_1 = (struct_local_data_storage_VTABLE_4 *)ILocalDataStorage::`vftable';
  v33 = a2;
  this->vftbl_0 = (struct_local_data_storage_VTABLE_0 *)LocalStorage::`vftable';
  this->vftbl_1 = (struct_local_data_storage_VTABLE_4 *)LocalStorage::`vftable'{for `ILocalDataStorage'};
  this->field_4 = 7;
  this->field_3 = 0;
```



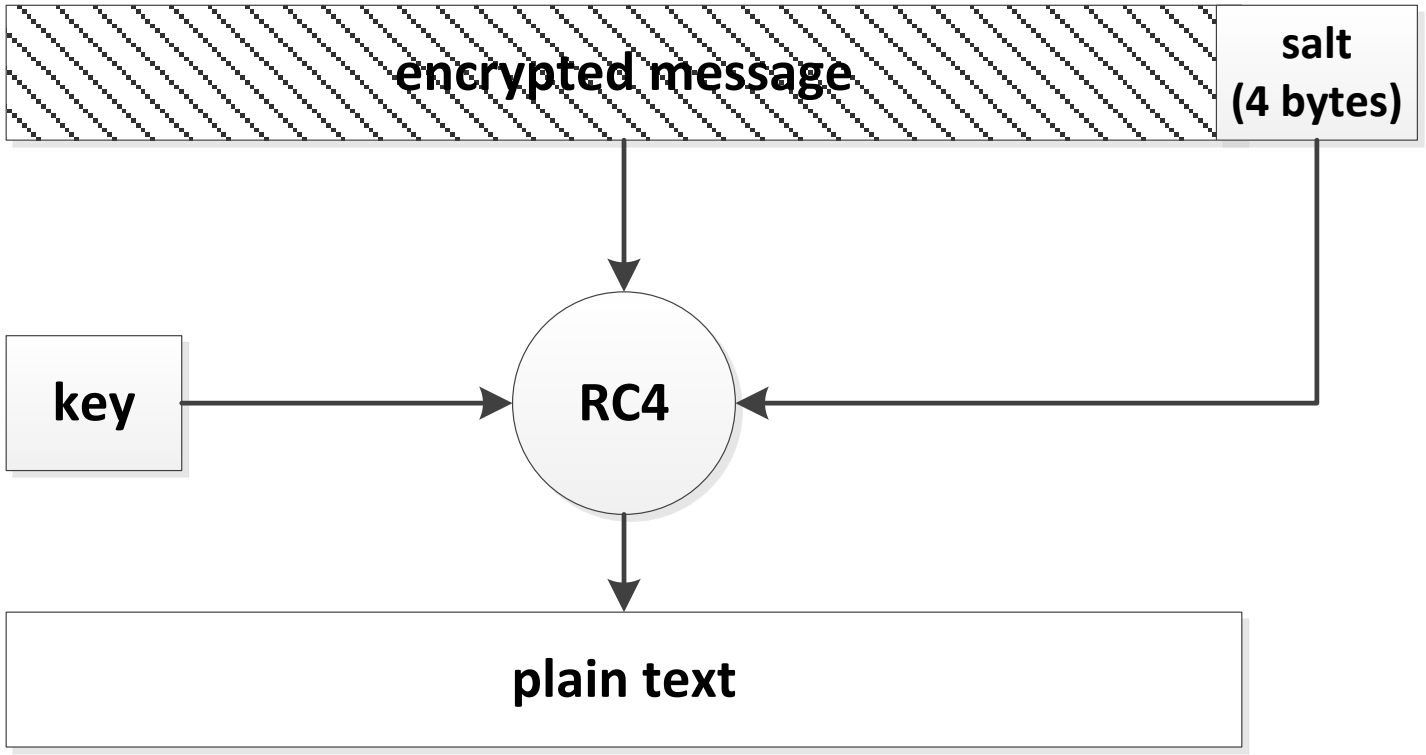
XAgent: Cryptor

```
100308FC 3          Cryptor                    Cryptor: ICryptor;
```

```
struct_cryptor *__thiscall Cryptor_Init(struct_cryptor *this, void *key, rsize_t key_size)
{
  struct_cryptor *v3; // esi@1
  struct_crypto_1 *v4; // eax@1
  struct_crypto_1 *v5; // eax@2

  v3 = this;
  this->vftbl_0 = (struct_cryptor_VTABLE_0 *)Cryptor::`vftable';
  this->field_1 = 0;
  v4 = (struct_crypto_1 *)operator new(8u);
  if ( v4 )
    v5 = init_buffer(v4, key, key_size);
  else
    v5 = 0;
  v3->key = v5;
  return v3;
}
```

```
.rdata:100308F8 ; class Cryptor: ICryptor;   (#classinformer)
.rdata:100308F8                 dd offset ??_R4Cryptor@@6B@ ; const Cryptor::`RTTI Complete Object Locator'
.rdata:100308FC ; const Cryptor::`vftable'
.rdata:100308FC ??_7Cryptor@@6B@ dd offset validate_buffer
.rdata:100308FC
.rdata:10030900                 dd offset encrypt_buffer_0
.rdata:10030904                 dd offset decrypt_buffer
.rdata:10030908
```

XAgent: Cryptor



XAgent: IReservedApi

```
10030954 3          ReservedApi                    ReservedApi: IReservedApi;
```

```
struct_name_node *__thiscall DNameNode::DNameNode(struct_name_node *this)
{
  struct_name_node *result; // eax@1

  result = this;
  this->vftbl_0 = (struct_name_node_VTABLE_0 *)&ReservedApi::`vftable';
  this->hMutex = 0;
  return result;
}
```

```
.rdata:10030950 ; class ReservedApi: IReservedApi;   (#classinformer)
.rdata:10030950                 dd offset ??_R4ReservedApi@@6B@ ; const ReservedApi::`RTTI Complete Object Locator'
.rdata:10030954 ; const ReservedApi::`vftable'
.rdata:10030954 ??_7ReservedApi@@6B@ dd offset get_volume_serial_number
.rdata:10030954
.rdata:10030958                 dd offset create_mutex
.rdata:1003095C                 dd offset shell_execute_open
```

XAgent: Identifying Used Types

* Strings: **std::string**

* Containers to maintain objects:
    -- **std::vector**
    -- **std::list**

XAgent: Identifying Used Types

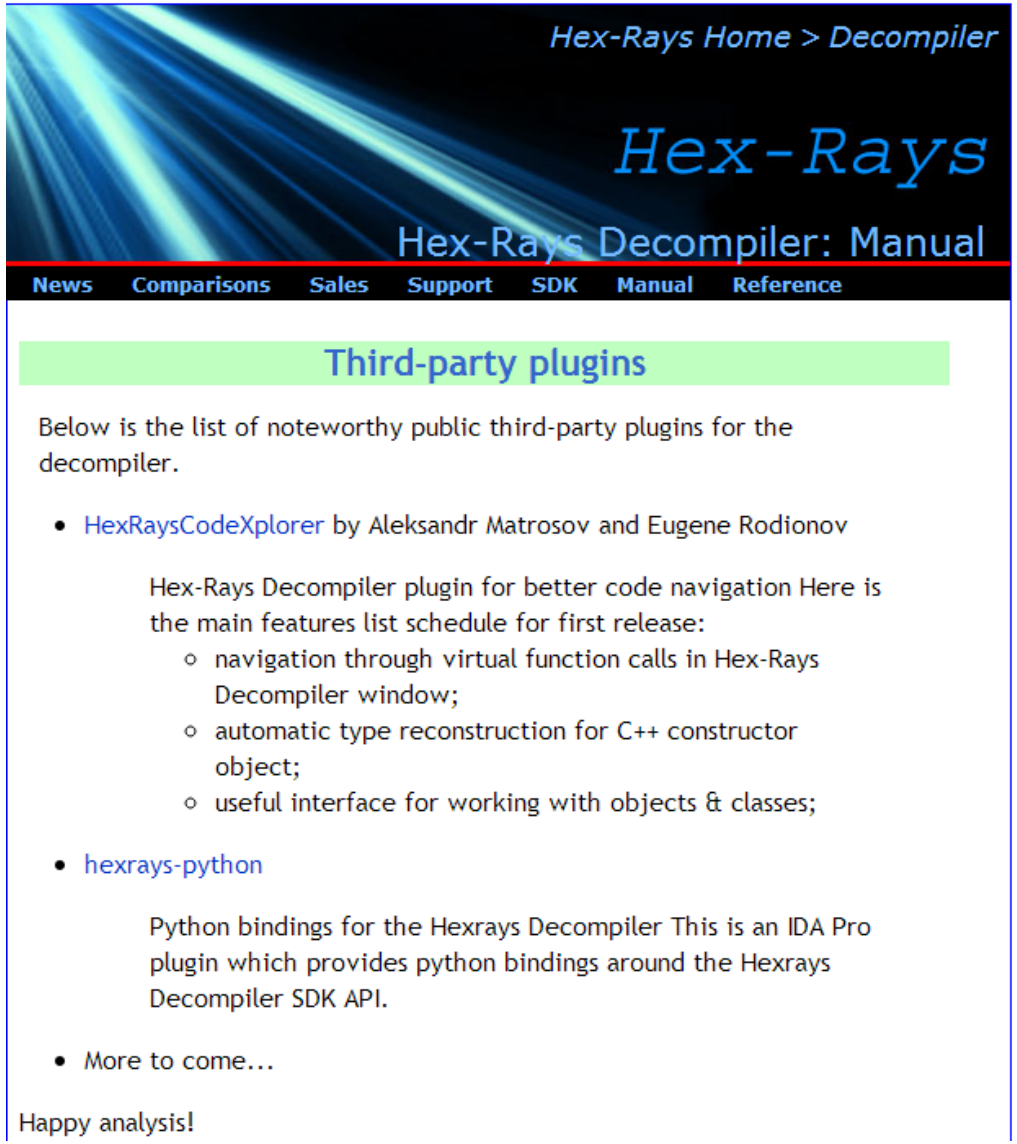
```
void *__thiscall std::string(void *this, int a2, unsigned int a3, unsigned int a4)
{
  v4 = this;
  v5 = a2;
  v6 = *(_DWORD *)(a2 + 16);
  if ( v6 < a3 )
    sub_100198C1("invalid string position");
  v7 = v6 - a3;
  if ( a4 < v7 )
    v7 = a4;
  if ( v4 == (void *)a2 )
  {
    sub_100014E0(a3 + v7, -1);
    sub_100014E0(0, a3);
    return v4;
  }
  if ( v7 > 0x7FFFFFFE )
    sub_10019874("string too long");
  v9 = *((_DWORD *)v4 + 5);
```

```
__declspec(noreturn) void _Xlen() const
    {   // report a length_error
    _Xlength_error("string too long");
    }

__declspec(noreturn) void _Xran() const
    {   // report an out_of_range error
    _Xout_of_range("invalid string position");
    }
};
```

* Containers to maintain objects:
  * -- **std::vector**
  * -- **std::list**

HexRaysCodeXplorer

HexRaysCodeXplorer since 2013



* CodeXplorer V1.0 released on REcon'2013

* First third-party plugin for Hex-Rays Decompiler

* v1.0 supports IDA v6.4 and Decompiler for x86 v1.8

HexRaysCodeXplorer Features

* Hex-Rays decompiler plugin x86/x64

* The plugin was designed to facilitate static analysis of:
  -- object oriented code
  -- position independent code

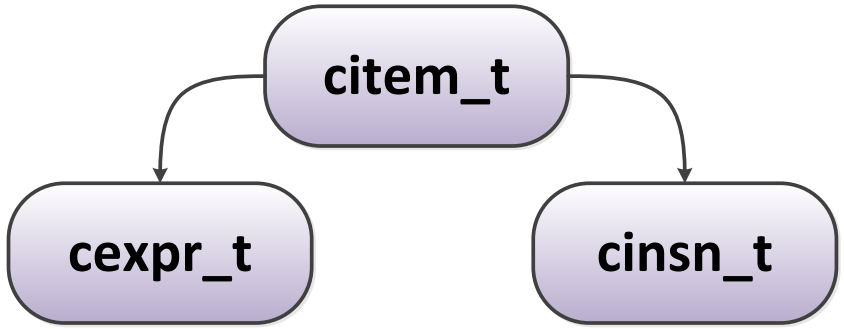
* The plugin allows to:
  -- partially reconstruct object type
  -- navigate through decompiled virtual methods

Hex-Rays Decompiler Plugin SDK

```
/// Ctree maturity level. The level will increase
/// as we switch from one phase of ctree generation to the next one
enum ctree_maturity_t
{
  CMAT_ZERO,            ///< does not exist
  CMAT_BUILT,           ///< just generated
  CMAT_TRANS1,          ///< applied first wave of transformations
  CMAT_NICE,            ///< nicefied expressions
  CMAT_TRANS2,          ///< applied second wave of transformations
  CMAT_CPA,             ///< corrected pointer arithmetic
  CMAT_TRANS3,          ///< applied third wave of transformations
  CMAT_CASTED,          ///< added necessary casts
  CMAT_FINAL,           ///< ready-to-use
};
```

nsec

Hex-Rays Decompiler Plugin SDK



* Type *citem_t* is a base class for:
  -- *cexpr_t* – expression type
  -- *cinsn_t* – statement type

* Expressions have attached type information

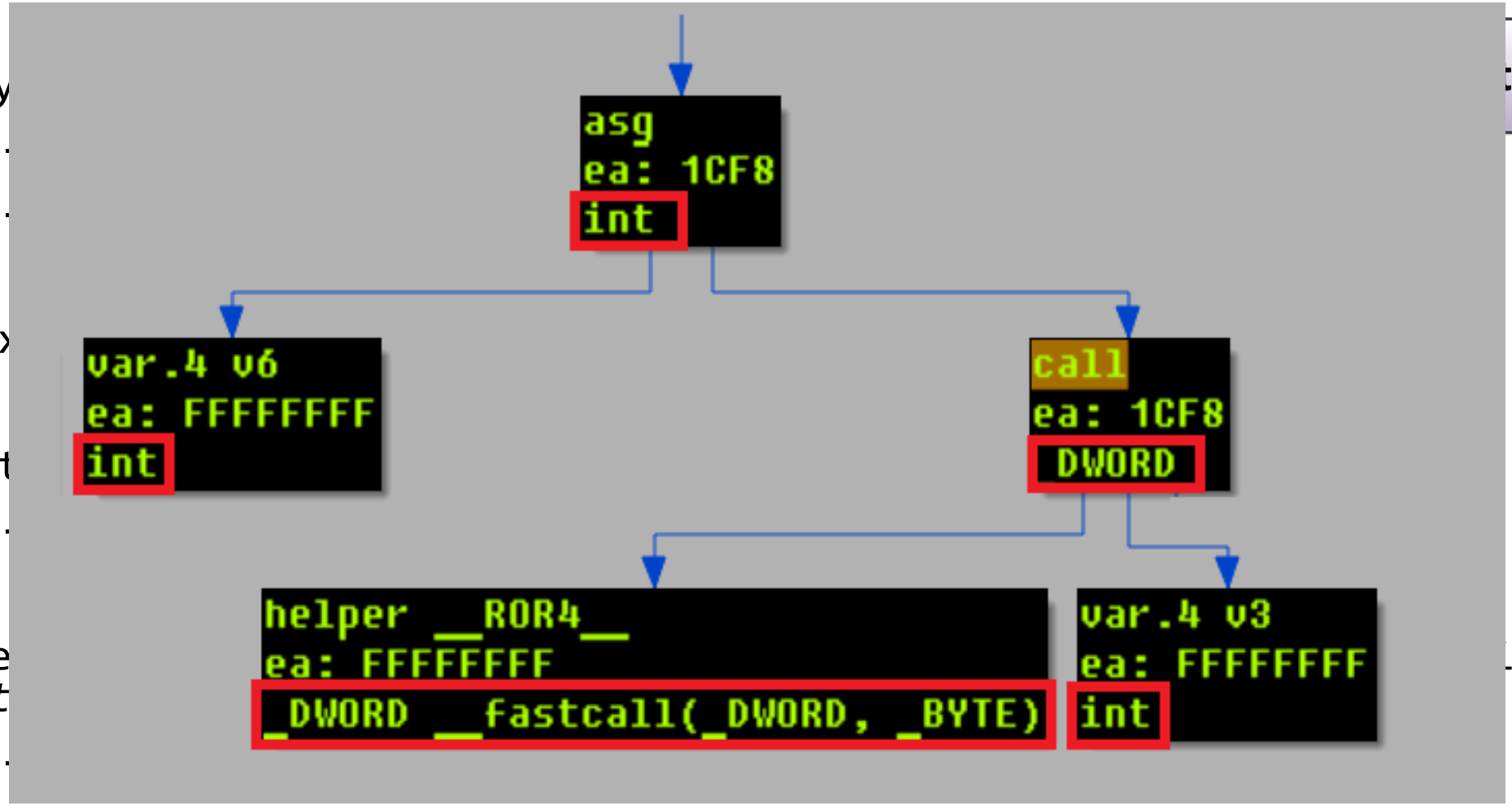
* Statements include:
  -- block, if, for, while, do, switch, return, goto, asm

* Hex-Rays provides iterators for traversing the *citem_t* objects within *ctree* structure:
  -- ctree_visitor_t, ctree_parentee_t

nsec

Hex-Rays Decompiler Plugin SDK

* Ty
* Ex
* St
* He
  ct



cinsn_t

thin

DEMO time :)

HexRaysCodeXplorer: Gapz Position Independent Code

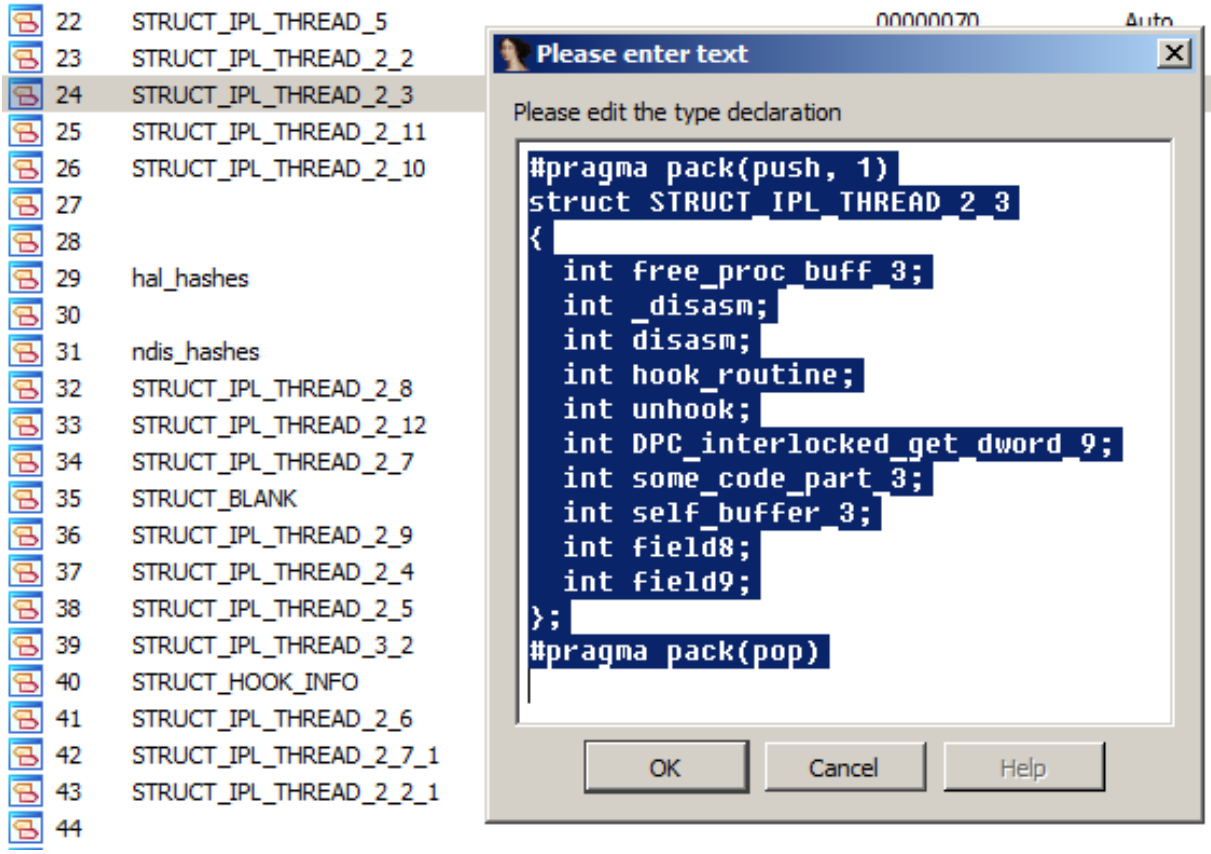
```
gl_context = (ExAllocatePoolWithTag)(0, 2576, 'ZPAG');
_gl_context = gl_context;
```

```
v12 = (get_export_by_hash)(kernel_base, hash_ntoskrnl_PsCreateSystemThread, v11);
v13 = hash_routin;
_gl_context->PsCreateSystemThread = v12;
v14 = (get_export_by_hash)(kernel_base, hash_ntoskrnl_PsTerminateSystemThread, v13);
v15 = hash_routin;
_gl_context->PsTerminateSystemThread = v14;
v16 = (get_export_by_hash)(kernel_base, hash_ntoskrnl_KeDelayExecutionThread, v15);
v17 = hash_routin;
_gl_context->KeDelayExecutionThread = v16;
```

```
_gl_context->ZwOpenSymbolicLinkObject)(&hSymLink, 0x80000000, &v301)
```

HexRaysCodeXplorer: Virtual Methods

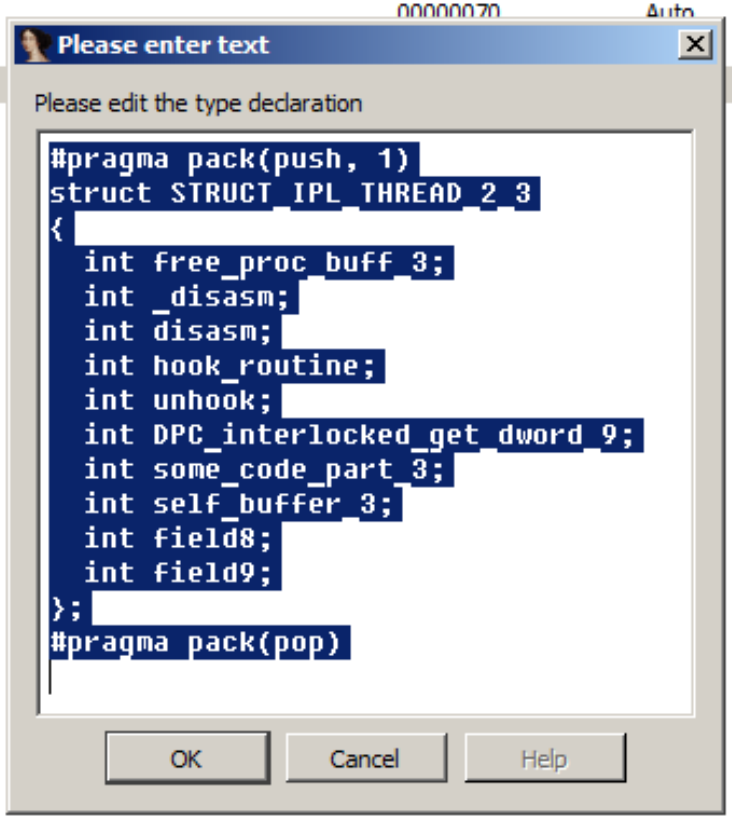
IDA's 'Local Types' is used to represent object type



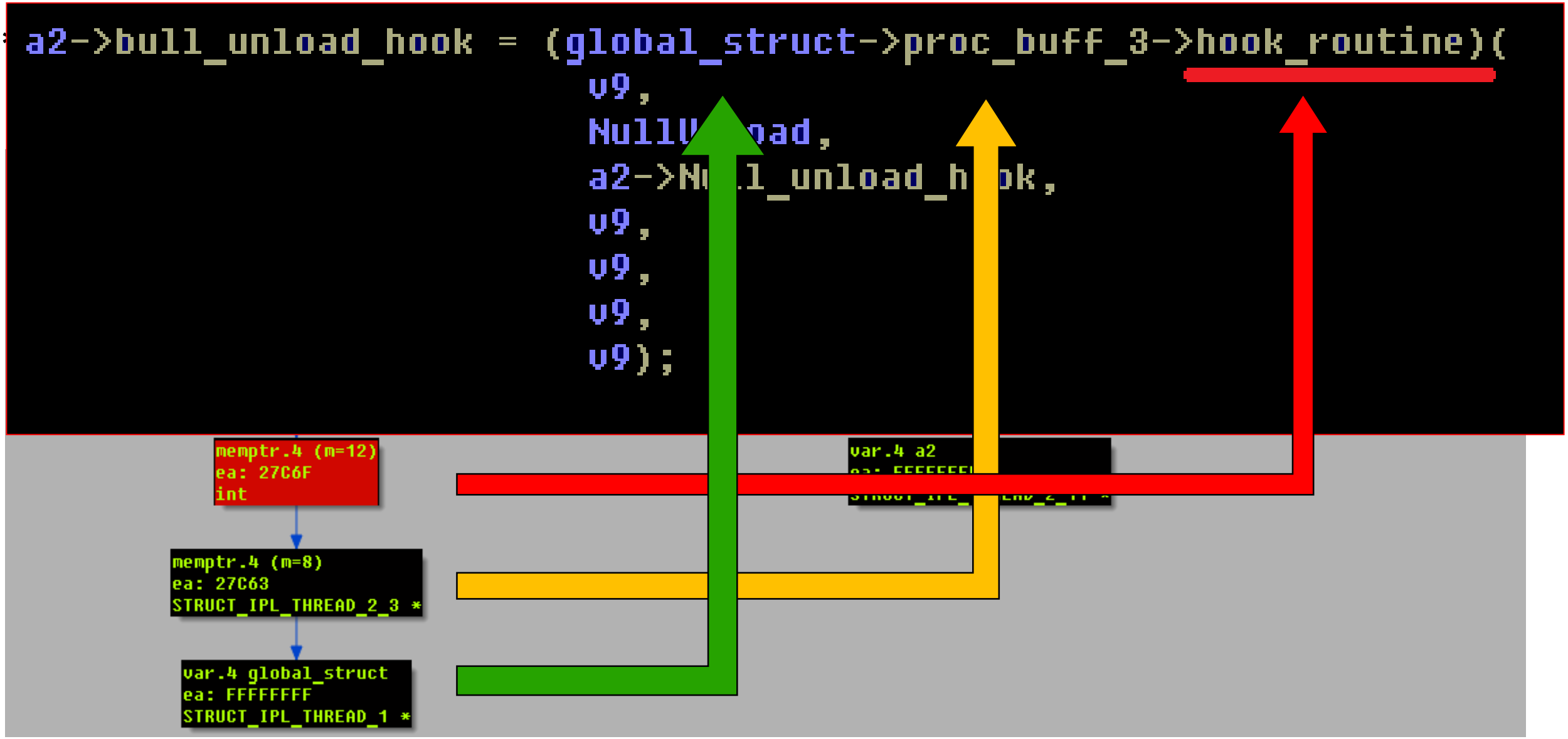
HexRaysCodeXplorer: Virtual Methods

```
int __stdcall block_3_init(STRUCT_IPL_THREAD_2_3 *self_buffer, STRUCT_IPL_THREAD_1 *a2)
{
  STRUCT_IPL_THREAD_2 *v2; // ebx@1
  int _self_buffer; // esi@1
  int (*get_some_code)(void); // edi@1
  STRUCT_IPL_THREAD_2_3 *v5; // eax@1
  int v6; // eax@1
  STRUCT_IPL_THREAD_1 *v7; // ST0C_4@1

  v2 = a2->proc_buffer;
  _self_buffer = self_buffer;
  get_some_code = (&self_buffer[0x36].field8 + -self_buffer->free_proc_buff_3 + 3);
  a2->proc_buffer->alloc_mem(a2->proc_buffer, &self_buffer, 40, 0);
  v5 = self_buffer;
  a2->proc_buff_3 = self_buffer;
  v5->self_buffer_3 = _self_buffer;
  self_buffer->free_proc_buff_3 = _self_buffer - *_self_buffer + 0x112F;
  self_buffer->DPC_interlocked_get_dword_9 = _self_buffer - *_self_buffer + 0xAA7;
  self_buffer->hook_routine = _self_buffer + 0xAF0 - *_self_buffer;
  self_buffer->unhook = _self_buffer + 0xF74 - *_self_buffer;
  self_buffer->_disasm = _self_buffer + 0x388 - *_self_buffer;
  self_buffer->disasm = self_buffer->_disasm;
  v6 = get_some_code();
  v7 = a2;
  self_buffer->some_code_part_3 = v6;           // D2B7
  (v2->replace_dword)(_self_buffer + 32, *(_self_buffer + 12), 0xBBBBBBBB, v7);
  return 0;
}
```

epresent



HexRaysCodeXplorer: Virtual Methods



HexRaysCodeXplorer: Object Type REconstruction

* Hex-Rays's *ctree* structure may be used to partially reconstruct object type

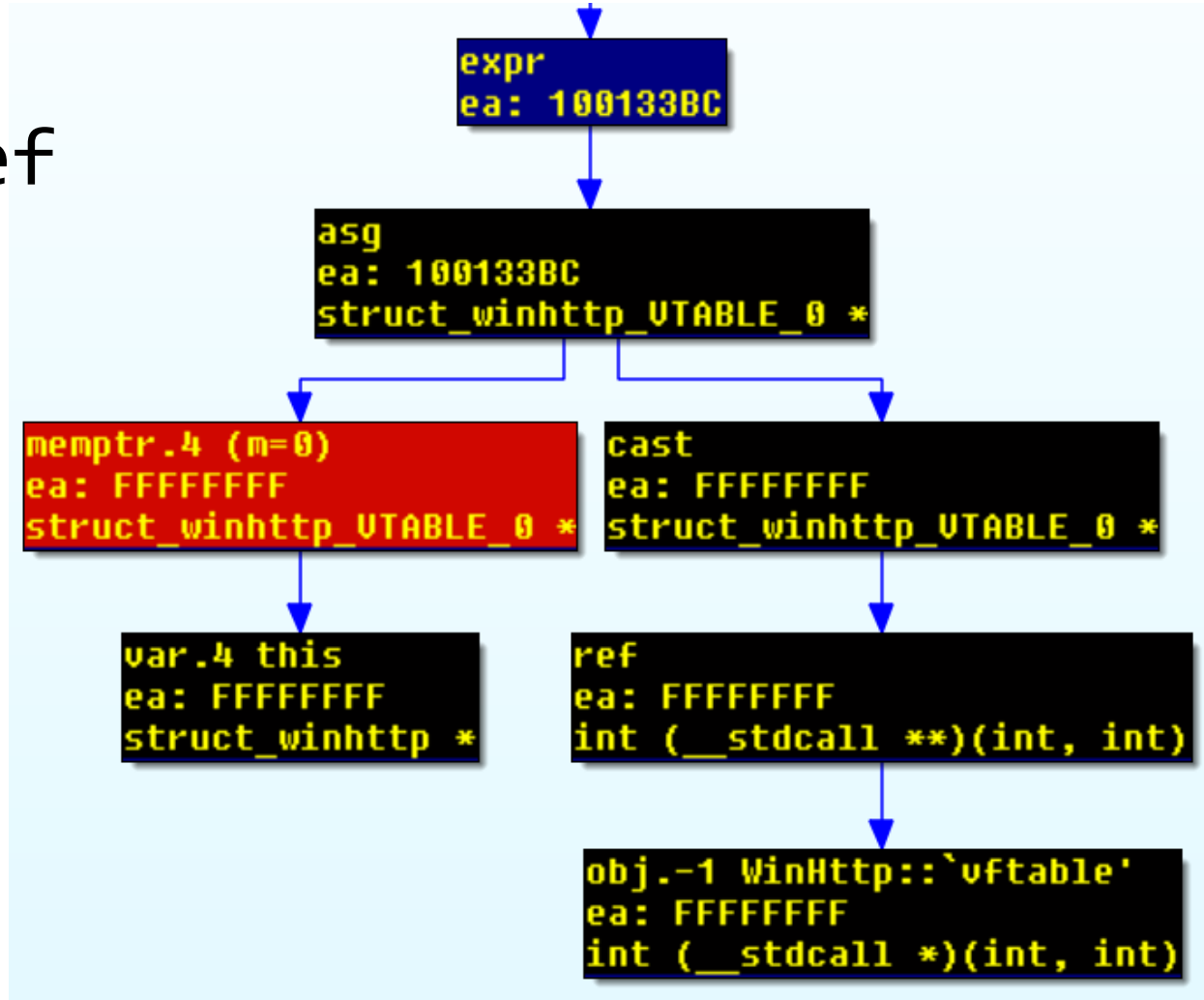
* Input:
  -- pointer to the object instance
  -- object initialization routine entry point

* Output:
  -- C structure-like object representation

HexRaysCodeXplorer: Object Type REconstruction

* *citem_t* objects:
    -- memptr, idx, memref
    -- call, ptr, asg

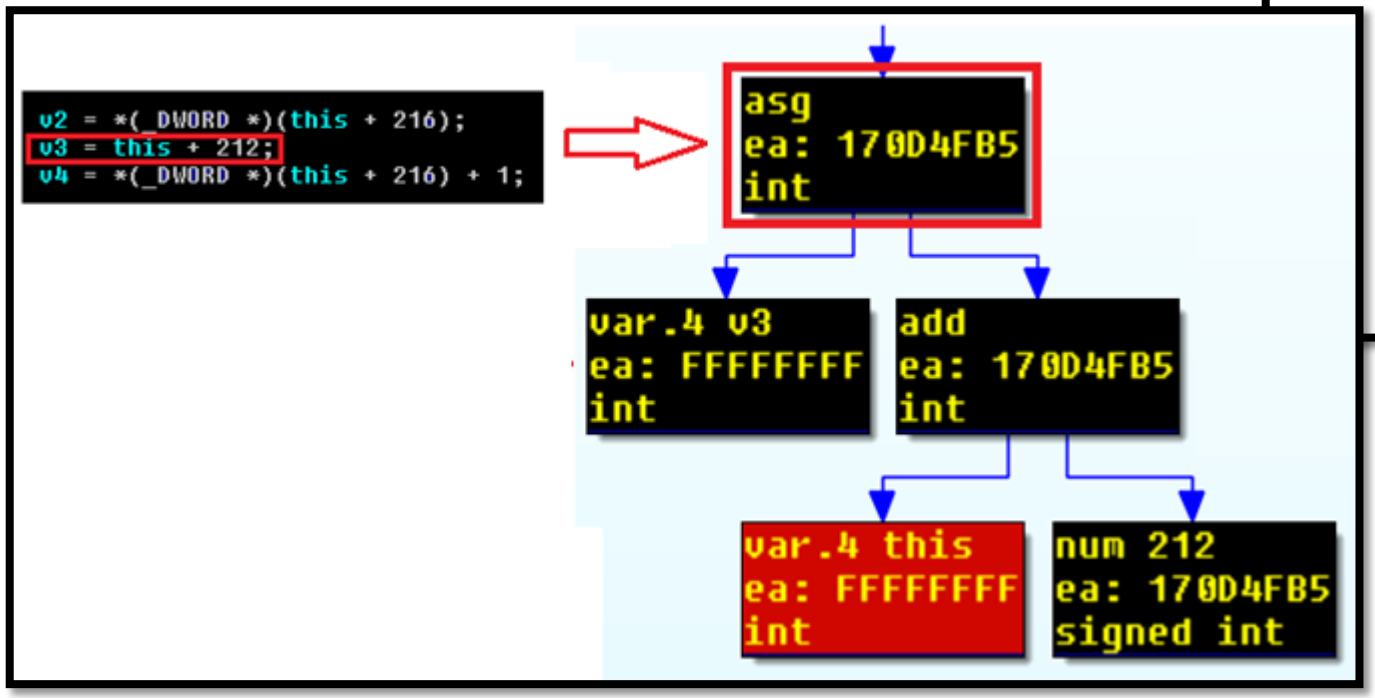
```
struct_winhttp *__thiscall WinHttp_Init(struct_winhttp *this)
{
  struct_winhttp *v1; // esi@1

  v1 = this;
  this->field_1 = 0;
  this->field_2 = 0;
  this->field_3 = 0;
  this->field_4 = 0;
  this->field_5 = 0;
  this->vftbl_0 = (struct_winhttp_VTABLE_0 *)&WinHttp::`vftable';
  this->field_10 = 7;
  this->field_9 = 0;
  this->ServerName = 0;
  this->field_12 = 0;
  this->field_11 = 0;
  this->field_7 = 15;
  this->field_6 = (int)calloc(0xFu, 1u);
  sub_100131F0(v1);
  sub_10011520();
  return v1;
}
```

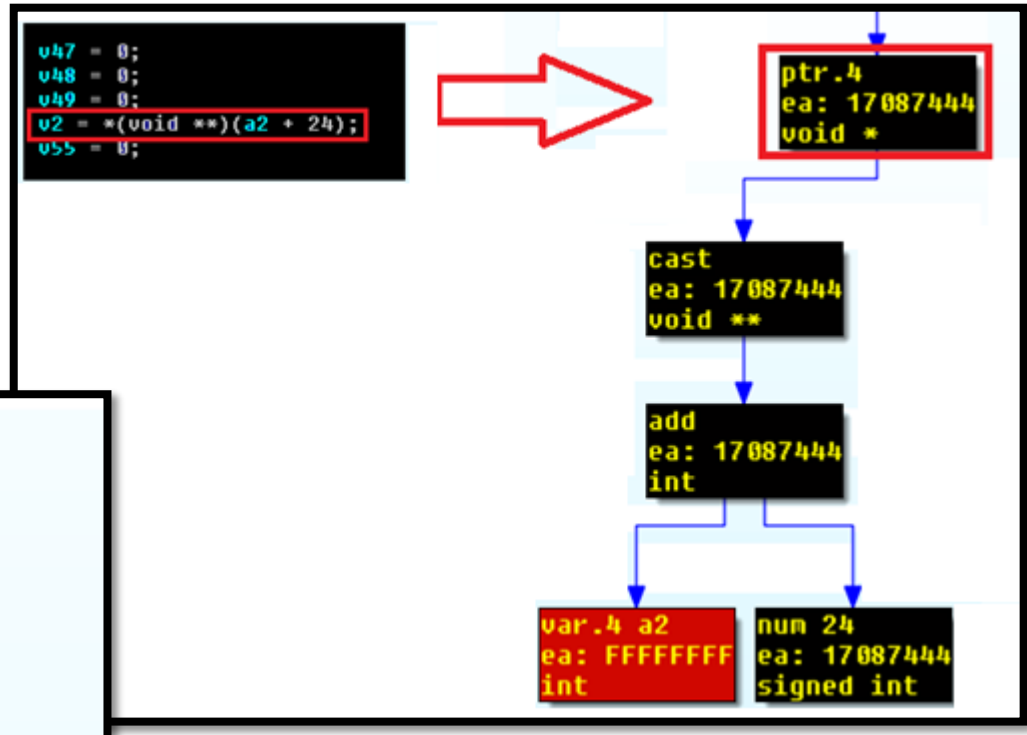


nsec

HexRaysCodeXplorer: Object Type REconstruction

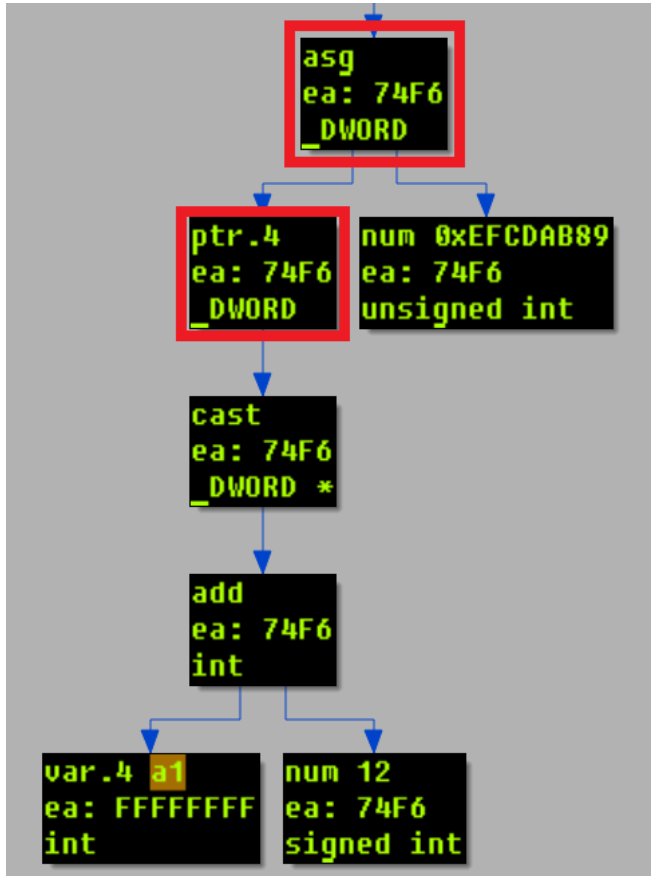
* *citem_t* objects:
    -- memptr, idx, memref
    -- call, **ptr, asg**





HexRaysCodeXplorer: Object Type REconstruction

```
// reference of DWORD at offset 12 in buffer a1
*(DWORD *)(a1 + 12) = 0xEFCDAB89;
```




```
17  this->field_4 = 7;
18  this->field_3 = 0;
19  this->field_2 = 0;
20  v2 = (char *)&this->field_5;
21  v10 = 0;
22  this->field_5 = 0;
23  v3 = operator new(8u);
24  if ( !v3 )
```

00007530 ModuleFileSystem:7

Output window

```
New type created:
struct
{
 struct_file_system_VTABLE_0 *vftbl_0;
 __int16 field_1;
 _BYTE gap6[6];
 __int16 field_2;
 _BYTE gapE[14];
 int field_3;
 int field_4;
 _BYTE gap24[8];
 int field_5;
}
```

HexRaysCodeXplorer: v1.7 [NSEC Edition]

Automatic virtual table identification

+

Type reconstruction

HexRaysCodeXplorer: v1.7 [NSEC Edition]

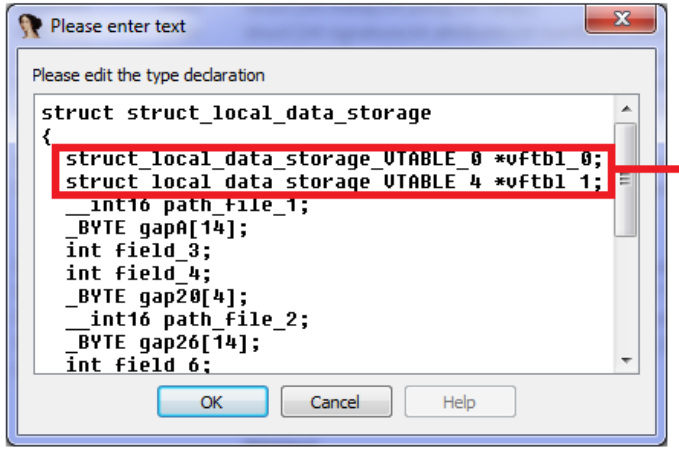
* **Automatic** virtual table identification

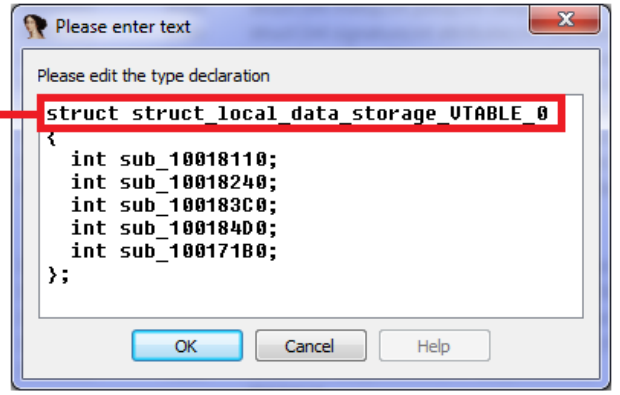
```
0x10030664 - 0x1003066c:  const std::system_error::`vftable'  methods count: 2
0x10030670 - 0x10030678:  const std::ios_base::failure::`vftable'  methods count: 2
0x100306c4 - 0x10030700:  const std::basic_streambuf<char,std::char_traits<char>>::`vftable'  methods count: 15
0x10030704 - 0x10030740:  const std::basic_stringbuf<char,std::char_traits<char>,std::allocator<char>>::`vftable'  methods count: 15
0x10030744 - 0x10030768:  const std::num_put<char,std::ostreambuf_iterator<char,std::char_traits<char>>>::`vftable'  methods count: 9
0x100307b4 - 0x100307cc:  const std::numpunct<char>::`vftable'  methods count: 6
0x10030840 - 0x10030854:  const ModuleRemoteKeyLogger::`vftable'  methods count: 5
0x10030858 - 0x10030860:  const IPTExternChannel::`vftable'  methods count: 2
0x10030864 - 0x1003086c:  const ProcessRetranslatorModule::`vftable'{for `IPTExternChannel'}  methods count: 2
0x10030870 - 0x10030884:  const ProcessRetranslatorModule::`vftable'  methods count: 5
0x10030894 - 0x100308b0:  const WinHttp::`vftable'  methods count: 7
0x100308b8 - 0x100308c8:  const std::tr1::_Ref_count<char>::`vftable'  methods count: 4
0x100308fc - 0x10030908:  const Cryptor::`vftable'  methods count: 3
0x1003090c - 0x1003091c:  const ILocalDataStorage::`vftable'  methods count: 4
0x10030920 - 0x10030930:  const LocalStorage::`vftable'{for `ILocalDataStorage'}  methods count: 4
0x10030934 - 0x10030948:  const LocalStorage::`vftable'  methods count: 5
0x10030954 - 0x10030960:  const ReservedApi::`vftable'  methods count: 3
0x10030980 - 0x10030998:  const CUnknown<IClassFactory>::`vftable'  methods count: 6
0x1003099c - 0x100309b4:  const CUnknown<IObjectWithSite>::`vftable'  methods count: 6
```

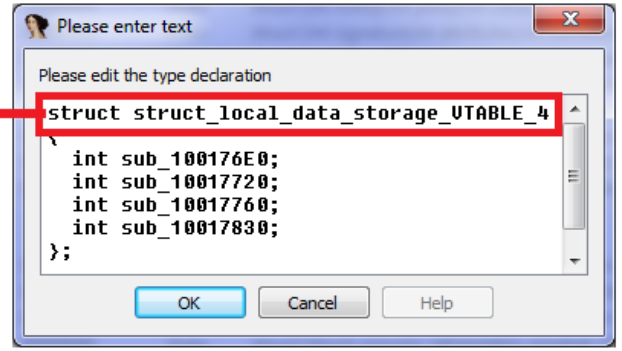
HexRaysCodeXplorer: v1.7 [NSEC Edition]

```
struct_local_data_storage *__thiscall LocalDataStorage_Init(struct_local_data_storage *this, void *a2, void *a3, int a4, int a5, int a6, int a7)
{
  v7 = this;
  this->vftbl_1 = (struct_local_data_storage_VTABLE_4 *)ILocalDataStorage::`vftable';
  v33 = a2;
  this->vftbl_0 = (struct_local_data_storage_VTABLE_0 *)LocalStorage::`vftable';
  this->vftbl_1 = (struct_local_data_storage_VTABLE_4 *)LocalStorage::`vftable'{for `ILocalDataStorage'};
  this->field_4 = 7;
  this->field_3 = 0;
```

```
0x10030664 - 0x1003066c:  const std::system_e
0x10030670 - 0x10030678:  const std::ios_base
0x100306c4 - 0x10030700:  const std::basic_st
0x10030704 - 0x10030740:  const std::basic_st
0x10030744 - 0x10030768:  const std::num_put<
0x100307b4 - 0x100307cc:  const std::numpunct
0x10030840 - 0x10030854:  const ModuleRemoteK
0x10030858 - 0x10030860:  const IPTExternChan
0x10030864 - 0x1003086c:  const ProcessRetran
0x10030870 - 0x10030884:  const ProcessRetran
0x10030894 - 0x100308b0:  const WinHttp::`vft
0x100308b8 - 0x100308c8:  const std::tr1::_Re
0x100308fc - 0x10030908:  const Cryptor::`vft
0x1003090c - 0x1003091c:  const ILocalDataSto
0x10030920 - 0x10030930:  const LocalStorage:
0x10030934 - 0x10030948:  const LocalStorage:
0x10030954 - 0x10030960:  const ReservedApi::
0x10030980 - 0x10030998:  const CUnknown<ICla
0x1003099c - 0x100309b4:  const CUnknown<IObj
```








```
count: 15
s count: 9
```

nsec

HexRaysCodeXplorer: v1.7 [NSEC Edition]

* **Automatic** virtual table identification

* Support for IDA Pro x64

* Bugfixes

DEMO time :)

HexRaysCodeXplorer: Next plans

* Switch to IdaPython

Why python?

```python
import idaapi

class CTreeVisitor(idaapi.ctree_visitor_t):
  def __init__(self, dumper, cfunc):
    idaapi.ctree_visitor_t.__init__(self, idaapi.CV_FAST | idaapi.CV_INSNS)
    self.dumper = dumper
    self.cfunc = cfunc
    return

  def visit_insn(self, ins):
      print ins.opname
      return 0

class CDumper(object):
  def __init__(self):
    self.ret = {}

  def dump(self, ea):
    f = idaapi.get_func(ea)
    cfunc = idaapi.decompile(f)
    visitor = CTreeVisitor(self, cfunc)
    visitor.apply_to(cfunc.body, None)

def main():
  dump = CAstDumper()
  dump.dump(here())

if __name__ == "__main__":
  main()
```

```
block
expr
expr
if
block
expr
expr
block
expr
expr
expr
expr
expr
expr
expr
expr
expr
if
block
expr
expr
block
expr
expr
expr
expr
expr
expr
expr
expr
expr
expr
return
```

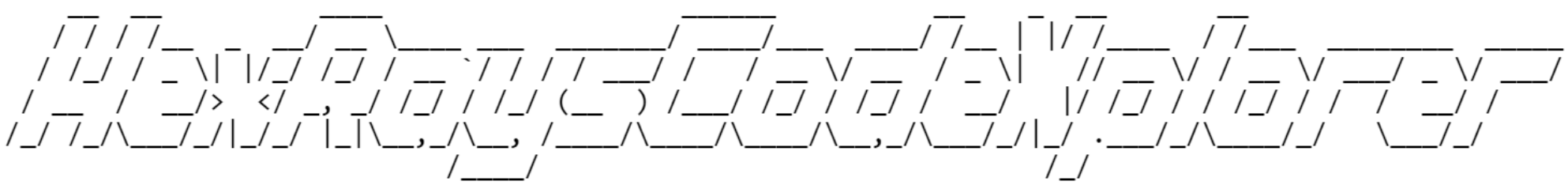
nsec

HexRaysCodeXplorer: Next plans

* Switch to IdaPython

* Further research & development:
  -- find cross-references to object attributes
  -- handling nested structures
  -- code similarity based on data flow analysis

Thank you for your attention!



http://REhints.com

@Rehints

https://github.com/REhints/HexRaysCodeXplorer