

What Happens In Windows 7 Stays In Windows 7

Joseph Moti & Marion Marschalek

POC2013

Agenda

- Vulnerabilities
- Automated Vulnerability Search
- An Approach
- A Solution as Proof of Concept
- Demo ;)
- Whats next?

Who are we

Joseph Moti

Moti Joseph has been involved in computer security. In the last few years he has been working on reverse engineering exploit code and developing security products.

Moti is a former speaker at Black Hat 2007, USA, CONF2009, Poland Warsaw, POC 2009, South Korea, ShakaCon 2009, USA, CONF2010, Poland Karkow, POC 2010, South Korea, CHINA 2011 at Shanghai Jiao Tong University , NOPcon 2012 and SysCan2010 Taiwan,Taipe

Marion Marschalek

Marion Marschalek is currently employed at IKARUS Security Software GmbH based in Vienna, Austria. She is working as Malware Analyst and in Incident Response. Marion was a former speaker at Defcon21 in Las Vegas and Hack.lu2013 in Luxembourg.

In March this year Marion won the Female Reverse Engineering Challenge 2013, organized by RE professional Halvar Flake.

Intro

Got a bug in your software?



Can I haz it??

Schneier on security.

Vulnerabilities are **software mistakes** in specification and design, but mostly mistakes **in programming**. Any large software package will have thousands of mistakes. Once discovered, they can be **used to attack systems**. This is the point of security patching: eliminating known vulnerabilities. But many systems don't get patched, so the **Internet is filled with known, exploitable vulnerabilities**.

How to find vulnerabilities?

- Application Penetration Testing
- Fuzzing
- Reverse Engineering
- Source Code Review
- Or.. Being more advanced:
 - Tracking software bugs, introducing bugs into software, reversing security patches

Who is interested in finding them?

- Hackers – its fun to break stuff, usually
- Software Companies – should find them (and fix them..)
- Criminals – use them for illegal stuff
- Governments – use them for semi-legal stuff
- Media – gets good news out of (exploited) vulnerabilities

And why automate it?

- It's faster!!
 - The hacker – can break more
 - The software company – can fix faster
 - Criminals – can make more money
 - Governments – can ... [SECRET]
 - Media – has more to write about

The Approach

What happens in Windows7 stays in Windows7...

Win7

quartz.dll

```
xor    eax, eax
inc    eax
shl    eax, cl
...
shl    eax, 2
push  eax        ; cb
call   ds:__imp__CoTaskMemAlloc@4
```

Patch it!

Win8

quartz.dll

```
lea    ecx, [ebp+cb]
push  ecx
push  4
push  eax
mov   [esi], eax
call  ?ULongMult@@YGJKKPAK@Z
test  eax, eax
...
push  [ebp+cb]        ; cb
call  ds:__imp__CoTaskMemAlloc@4
```

Counting Function Calls

Win7 quartz.lib

Address	Function	Ins
.text:76039427	?ULongAdd@@YGJKPAK@Z	
.text:76131235	?NotifyExternalMemory@CRe...	
.text:76130CC7	?Configure@CRecCache@@@...	
.text:76130C44	?Configure@CRecCache@@@...	
.text:7612F39D	??0CImplReader_1@@@QAE@P...	
.text:7612F387	??0CImplReader_1@@@QAE@P...	
.text:7612F346	??0CImplReader_1@@@QAE@P...	
.text:7612F2DD	??0CImplReader_1@@@QAE@P...	
.text:7612E938	?AlignUp@CImplReader_1@...	
.text:7612B58B	?CopyImage@CBaseControlVi...	
.text:76115480	?GetFrame@CID3Parse@@CG...	
.text:76115438	?GetFrame@CID3Parse@@CG...	
.text:761153B0	?ExtendedHeaderLength@CID...	
.text:7610981E	?WSTRFromAnsi@@YGJPAPA...	
.text:761078C7	sub_76107849	
.text:76104E48	?CreateOutputPins@CWAVEP...	
.text:76104DF4	?CreateOutputPins@CWAVEP...	
.text:76104DDC	?CreateOutputPins@CWAVEP...	
.text:76104C2E	?CreateOutputPins@CWAVEP...	
.text:76104C19	?CreateOutputPins@CWAVEP...	
.text:76104BBA	?CreateOutputPins@CWAVEP...	
.text:761039E5	??0CImplOldAviIndex@@@QAE...	
.text:761035A5	?ValidateSuperIndex@CImplSt...	
.text:7610351A	?ValidateStdIndex@CImplStdA...	
.text:7610179A	?BuildMT@CAviMSROutPin@...	
.text:761004DA	?SearchList@@YGJPAUIAsync...	
.text:76100421	?Search@CAviMSRFilter@@@A...	
Line 35 of 44		

Address	Function	Instruction
.text:35532BA1	_ConvertVideoInfoToVideoInf...	call ?ULongAdd@@YGJK
.text:35620869	??0CImplReader_1@@@QAE@P...	call ?ULongAdd@@YGJK
.text:356208F0	??0CImplReader_1@@@QAE@P...	call ?ULongAdd@@YGJK
.text:35620930	??0CImplReader_1@@@QAE@P...	call ?ULongAdd@@YGJK
.text:355335DD	_CheckMPEG1VideoInfoType@4	call ?ULongAdd@@YGJK
.text:3557CC3B	?GetMaxConnectionsSize@CF...	call ?ULongAdd@@YGJK
.text:3562094C	??0CImplReader_1@@@QAE@P...	call ?ULongAdd@@YGJK
.text:3554D675	_CheckMPEG2VideoInfoType@4	call ?ULongAdd@@YGJK
.text:355772FB	?CopyImage@CBaseControlVi...	call ?ULongAdd@@YGJK
.text:3557CC28	?GetMaxConnectionsSize@CF...	call ?ULongAdd@@YGJK
.text:356210E6	sub_356210D4	call ?ULongAdd@@YGJK
.text:3557CC4C	?GetMaxConnectionsSize@CF...	call ?ULongAdd@@YGJK
.text:3557CC8A	?GetMaxConnectionsSize@CF...	call ?ULongAdd@@YGJK
.text:3557CCD3	?GetMaxConnectionsSize@CF...	call ?ULongAdd@@YGJK
.text:35582C5D	?MediaTypeToText@@YGJVC...	call ?ULongAdd@@YGJK
.text:35592000	?GetMediaType@CAVIDec@@@...	call ?ULongAdd@@YGJK
.text:3559300C	?StartStreaming@CMJPGEnc...	call ?ULongAdd@@YGJK
.text:35593B95	?Transform@CMJPGEnc@@@U...	call ?ULongAdd@@YGJK
.text:355CBD7C	?CopyRGBSurfToDIB@Calloca...	call ?ULongAdd@@YGJK
.text:355D45C8	?SetToVideoInfoHeader2@VP...	call ?ULongAdd@@YGJK
.text:355EC6F1	?CopyRGBSurfToDIB@Calloca...	call ?ULongAdd@@YGJK
.text:355FF06F	?BuildMT@CAviMSROutPin@...	call ?ULongAdd@@YGJK
.text:355FF87F	?SearchList@@YGJPAUIAsync...	call ?ULongAdd@@YGJK
.text:355FF95D	?Search@CAviMSRFilter@@@A...	call ?ULongAdd@@YGJK
.text:35600A68	?ValidateSuperIndex@CImplSt...	call ?ULongAdd@@YGJK
.text:35600ADB	?ValidateStdIndex@CImplStdA...	call ?ULongAdd@@YGJK
.text:3560126F	??0CImplOldAviIndex@@@QAE...	call ?ULongAdd@@YGJK
Line 45 of 46		

Win8 quartz.lib

Spot The Patch

Win7 quartz.lib

```
cmp     cx, 8
jbe     short loc_7609B8EC
mov     eax, 80040220h
jmp     short loc_7609B933

-----

; CODE XREF:
xor     eax, eax
inc     eax
shl     eax, cl
push   esi
mov     esi, [ebp+arg_0]
push   edi
mov     [esi], eax
shl     eax, 2
push   eax ; cb
call   ds:__imp__CoTaskMemAlloc@4 ;
mov     edi, [ebp+arg_4]
mov     [edi], eax
test    eax, eax
jnz    short loc_7609B914
and     [esi], eax
```

Win8 quartz.lib

```
lea     ecx, [ebp+cb]
push   ecx ; unsigned __int3
push   4 ; int
push   eax ; int
mov     [esi], eax
call   ?ULongMult@@YGJKKPAK@Z ; ULongMul
test    eax, eax
jns    short loc_355B131C
mov     eax, 80070216h
jmp     short loc_355B1351

-----

; CODE XREF: COve
push   [ebp+cb] ; cb
call   ds:__imp__CoTaskMemAlloc@4 ; CoTa
mov     [edi], eax
test    eax, eax
jnz    short loc_355B1334
and     [esi], eax
mov     eax, 8007000Eh
jmp     short loc_355B1351
```

Intsafe.h & Strsafe.h

- Searching for security patches:
 - Type Conversion
 - Safe Math Functions
 - Buffer Boundary Checks on Strings
- Set of 130 Signatures of ‚Safe Functions‘

,Safe Functions'

UInt8ToInt8

UInt8ToChar

ByteToInt8

ByteToChar

ShortToInt8

ShortToUChar

ShortToChar

UShortToUInt8

UShortToShort

IntToInt8

IntToUChar

IntToChar

UInt8Add

UShortAdd

UIntAdd

ULongAdd

SizeTAdd

ULongLongAdd

UInt8Sub

UShortSub

UIntSub

ULongSub

SizeTSub

ULongLongSub

StringCbGets

StringCbGetsEx

StringCbLength

StringCbPrintf

StringCbPrintfEx

StringCbVPrintf

StringCbVPrintfEx

StringCchCat

StringCchCatEx

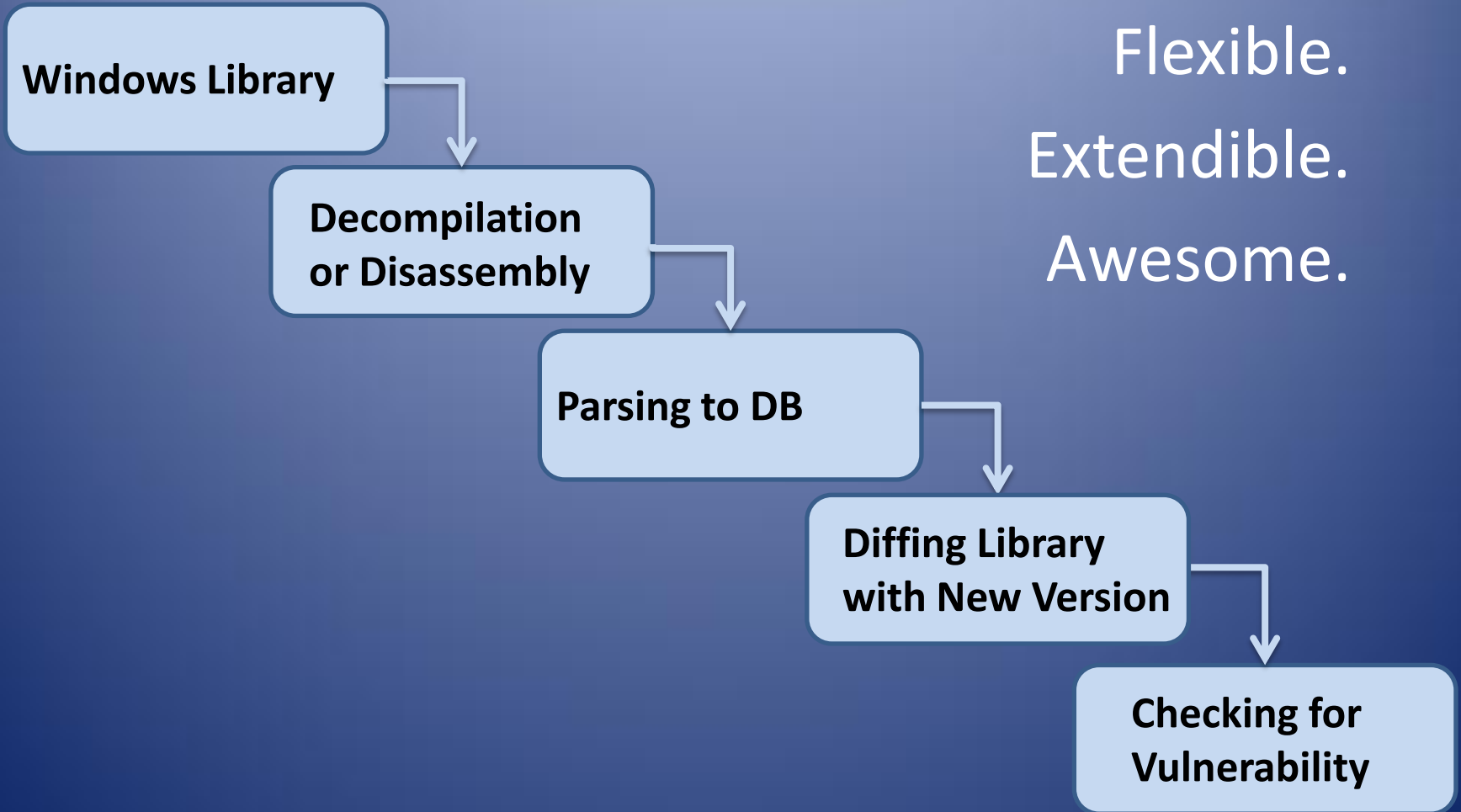
StringCchCatN

StringCchCatNEx

StringCchCopy

... and many many more

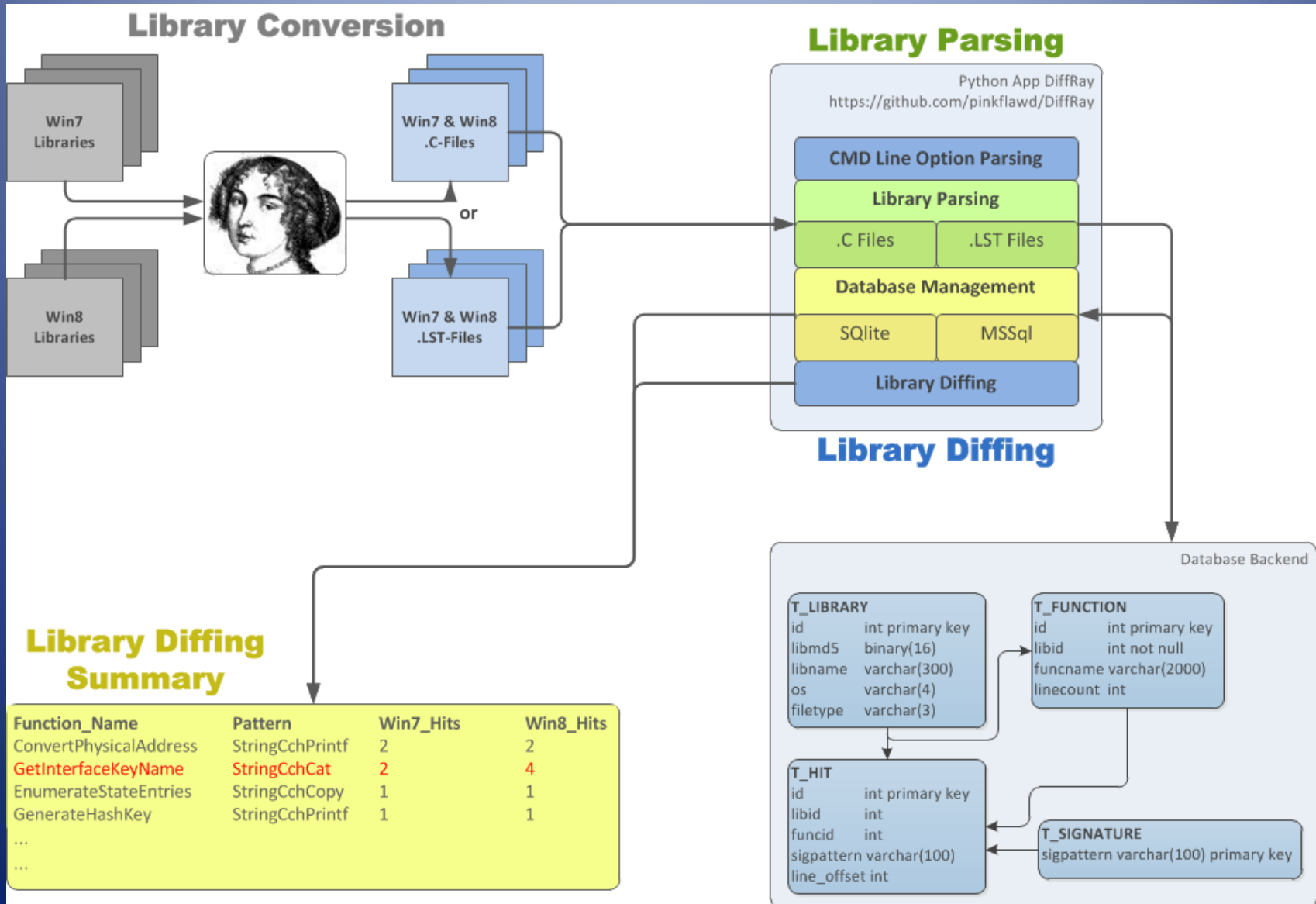
The Approach



Flexible.
Extendible.
Awesome.

The Solution

Pretty, eh?



Getting the .C

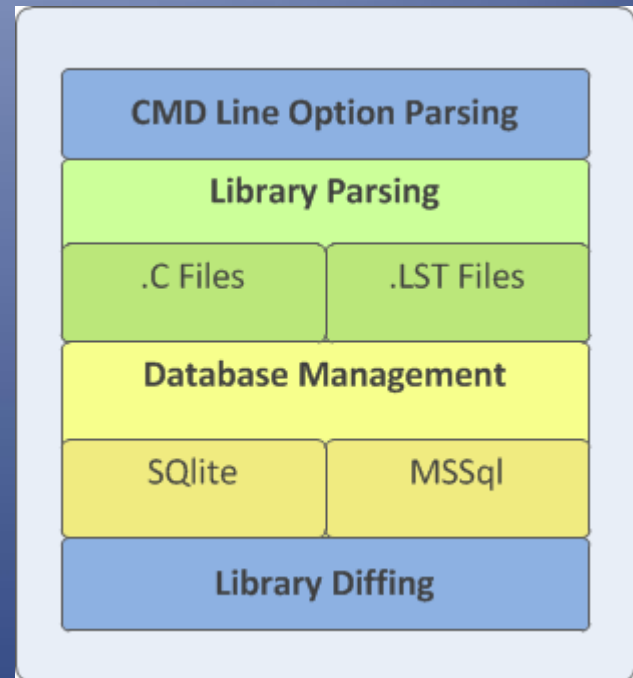
- Library Conversion using IDA Pro



- means: `.dll -> .idb -> .c`

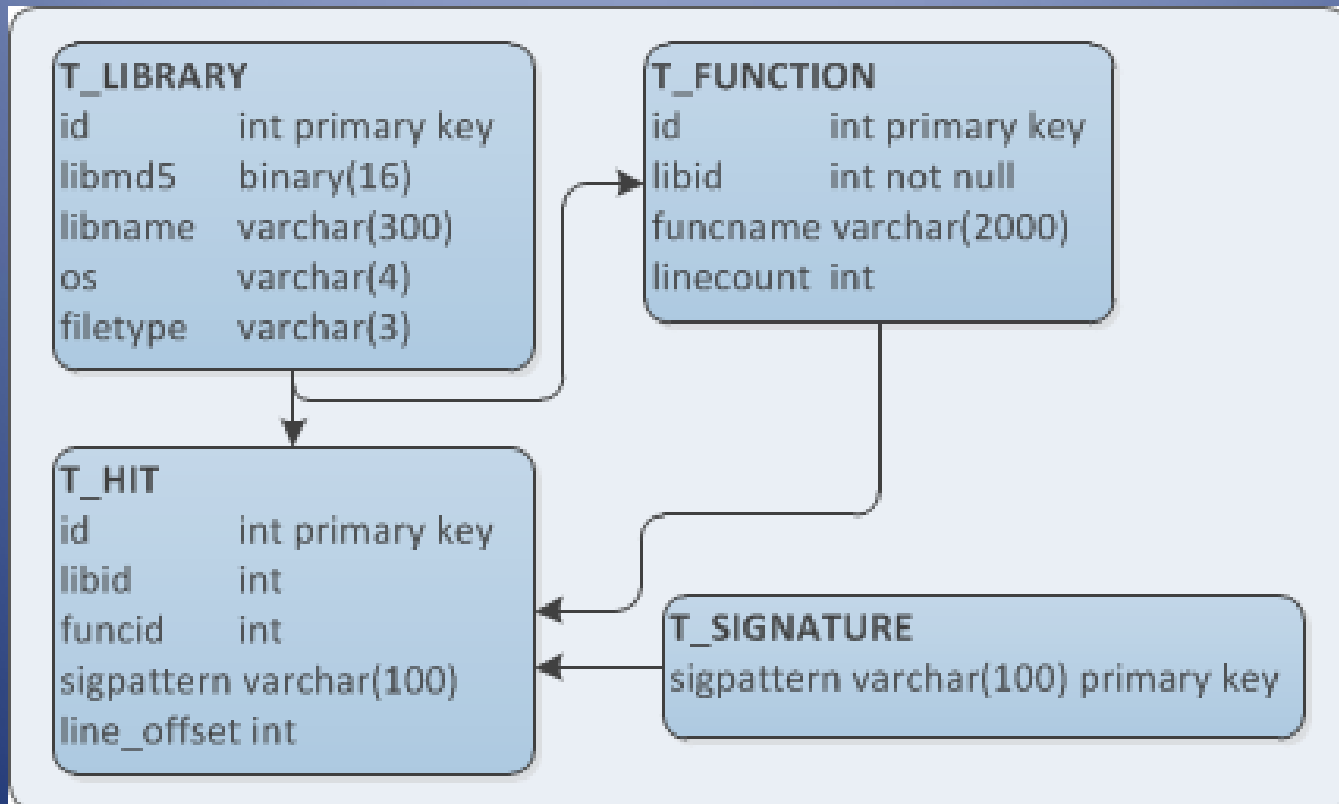
Library Parsing

- DiffRay on <https://github.com/pinkflawd/DiffRay>
- Parses a library / directory of libraries
- Manages libraries , functions and signature hits
- Diff libraries functionwise
 - Based on library ID or library name pattern



The Database

- MSSql or SQLite (for now)



Diff it!

- Compare libraries on a function basis
- Extract hits per function per signature

Function_Name	Pattern	Win7_Hits	Win8_Hits
ConvertPhysicalAddress	StringCchPrintf	2	2
GetInterfaceKeyName	StringCchCat	2	4
EnumerateStateEntries	StringCchCopy	1	1
GenerateHashKey	StringCchPrintf	1	1
...			
...			

DiffRay HowTo: Configuration

- **signatures.conf** – whatever symbols you're searching for
- **logger.conf** – logging output and formatting, details to be found at *<http://docs.python.org/2/howto/logging.html>*
- **mssql.conf** – MSSql access credentials

DiffRay HowTo: Parsing

Maintenance:

```
python [dir]\src\Main.py --create-scheme --update-sigs
```

```
python [dir]\src\Main.py --parse [library_path]  
                        --os [Win7|Win8] --type [C|LST]
```

```
python [dir]\src\Main.py --dirparse [directory_path]  
                        --os [Win7|Win8] --type [C|LST]
```

```
python [dir]\src\Main.py --flushall
```

Switches:

```
--backend [mssql|sqlite]
```

```
--no-flush
```


DiffRay HowTo: Diffing

Info Output & Diffing:

```
python [dir]\src\Main.py --search_libs [libname_pattern]
```

```
python [dir]\src\Main.py --lib_all_info [lib_id]
```

```
python [dir]\src\Main.py --diff  
                        --lib_1 [win7lib] --lib_2 [win8lib]
```

```
python [dir]\src\Main.py --diff_byname [libname_pattern]
```

Demo time!



Findings

Windows 8

bcrypt.dll!ConvertRsaPrivateBlobToFullRsa

```
Windows Help
No debugger
Hex View-A Structures Enums Imports Exports
.text:1000BFEE stosd
.text:1000BFEF mov [ebp+Dst], ecx
.text:1000BFF5 stosd
.text:1000BFF6 push 6
.text:1000BFF8 pop ecx
.text:1000BFF9 mov esi, edx
.text:1000BFFB lea edi, [ebp+var_F8]
.text:1000C001 rep movsd
.text:1000C003 mov ecx, [ebp+var_F0]
.text:1000C009 add ecx, 18h
.text:1000C00C mov [ebp+var_C4], edx
.text:1000C012 cmp ecx, 18h
.text:1000C015 jb loc_1000C281
.text:1000C01B mov esi, [ebp+var_EC]
.text:1000C021 lea eax, [esi+ecx]
.text:1000C024 cmp eax, ecx
.text:1000C026 jb loc_1000C281
.text:1000C02C mov ebx, [ebp+var_E8]
.text:1000C032 lea edx, [ebx+eax]
.text:1000C035 cmp edx, eax
.text:1000C037 jb loc_1000C281
.text:1000C03D mov ecx, [ebp+var_E4]
.text:1000C043 lea eax, [ecx+edx]
.text:1000C046 mov [ebp+var_DC], eax
.text:1000C04C cmp eax, edx
.text:1000C04E jb loc_1000C281
.text:1000C054 cmp [ebp+arg_0], eax
.text:1000C057 jnb short loc_1000C069
.text:1000C059 mov esi, 80090005h
.text:1000C05E push 626h
.text:1000C063 loc_1000C063: ; CODE XREF: ConvertRsaPrivateBlobToFullRsa(x,x,x,x,x)+16B↓j
.text:1000C063 push esi
.text:1000C064 jmp loc_1000C28E
```

Windows 7 (ULongAdd)

bcrypt.dll!ConvertRsaPrivateBlobToFullRsa

```
text:6D80C94E      push    18h
text:6D80C950      pop     eax
text:6D80C951      lea    edi, [ebp+var_2C]
text:6D80C954      rep    movsd
text:6D80C956      lea    ecx, [ebp+var_4]
text:6D80C959      push   ecx
text:6D80C95A      push   [ebp+var_24]
text:6D80C95D      mov    [ebp+var_4], eax
text:6D80C960      push   eax
text:6D80C961      call   _ULongAdd@12    ; ULongAdd(x,x,x)
text:6D80C966      test   eax, eax
text:6D80C968      jl     loc_6D80CB1E
text:6D80C96E      mov    esi, [ebp+var_20]
text:6D80C971      lea    eax, [ebp+var_4]
text:6D80C974      push   eax
text:6D80C975      push   esi
text:6D80C976      push   [ebp+var_4]
text:6D80C979      call   _ULongAdd@12    ; ULongAdd(x,x,x)
text:6D80C97E      test   eax, eax
text:6D80C980      jl     loc_6D80CB1E
text:6D80C986      mov    ebx, [ebp+var_1C]
text:6D80C989      lea    eax, [ebp+var_4]
text:6D80C98C      push   eax
text:6D80C98D      push   ebx
text:6D80C98E      push   [ebp+var_4]
text:6D80C991      call   _ULongAdd@12    ; ULongAdd(x,x,x)
text:6D80C996      test   eax, eax
text:6D80C998      jl     loc_6D80CB1E
text:6D80C99E      mov    edi, [ebp+var_18]
text:6D80C9A1      lea    eax, [ebp+var_4]
text:6D80C9A4      push   eax
text:6D80C9A5      push   edi
text:6D80C9A6      push   [ebp+var_4]
text:6D80C9A9      call   _ULongAdd@12    ; ULongAdd(x,x,x)
text:6D80C9AE      test   eax, eax
text:6D80C9B0      jl     loc_6D80CB1E
```

000BD4E|6D80C94E: ConvertRsaPrivateBlobToFullRsa(x,x,x,x,x)+14

Windows 7

netlogon.dll! NlpAddResourceGroupsToSamInfo

```
.\netlogon.dll
Debugger  Options  Windows  Help
[Icons] [No debugger]
IDA View-A  Pseudocode-A  Occurrences of: NlpAddResourceGroupsToSamInfo  Hex View-A  Structures  Enums
97  + *(_WORD *)(v3 + 80)
98  + 16;
99  if ( a1 == 6 )
100  v11 = ((((((((((((((((((((((((*(_WORD *)(v3 + 204) + v11 + *(_WORD *)(v3 + 212) + 5) & 0xFFFFFFFF)
101  + *(_WORD *)(v3 + 220)
102  + 3) & 0xFFFFFFFF)
103  + *(_WORD *)(v3 + 228)
104  + 3) & 0xFFFFFFFF)
105  + *(_WORD *)(v3 + 236)
106  + 3) & 0xFFFFFFFF)
107  + *(_WORD *)(v3 + 244)
108  + 3) & 0xFFFFFFFF)
109  + *(_WORD *)(v3 + 252)
110  + 3) & 0xFFFFFFFF)
111  + *(_WORD *)(v3 + 260)
112  + 3) & 0xFFFFFFFF)
113  + *(_WORD *)(v3 + 268)
114  + 3) & 0xFFFFFFFF)
115  + *(_WORD *)(v3 + 276)
116  + 3) & 0xFFFFFFFF)
117  + *(_WORD *)(v3 + 284)
118  + 3) & 0xFFFFFFFF)
119  + *(_WORD *)(v3 + 292)
120  + 2;
121  v33 = (v11 + 1) & 0xFFFFFFFF;
122  v12 = MIDL_user_allocate(v33);
123  v30 = v12;
124
```

Windows 8 (ULONGAdd)

netlogon.dll! NlpAddResourceGroupsToSamInfo

\syswin8\netlogon.dll

Options Windows Help

Code Data Comments Disassembly Hex View Pseudocode Occurrences of: NlpAddResourceGroupsToSamInfo No debugger

IDA View-A Pseudocode-A Occurrences of: NlpAddResourceGroupsToSamInfo Hex View-A Structures Enums Im

```
109 }
110 if ( a1 == 6 )
111 {
112     if ( RtlULONGAdd(size, *((_WORD *)v3 + 106) + *((_WORD *)v3 + 102) + 4, &size) < 0
113         || (v12 = 2, NetpULONGPtrRoundUp(size, 2, &size) < 0)
114         || RtlULONGAdd(size, *((_WORD *)v3 + 110) + 2, &size) < 0
115         || NetpULONGPtrRoundUp(size, 2, &size) < 0
116         || RtlULONGAdd(size, *((_WORD *)v3 + 114) + 2, &size) < 0
117         || NetpULONGPtrRoundUp(size, 2, &size) < 0
118         || RtlULONGAdd(size, *((_WORD *)v3 + 118) + 2, &size) < 0
119         || NetpULONGPtrRoundUp(size, 2, &size) < 0
120         || RtlULONGAdd(size, *((_WORD *)v3 + 122) + 2, &size) < 0
121         || NetpULONGPtrRoundUp(size, 2, &size) < 0
122         || RtlULONGAdd(size, *((_WORD *)v3 + 126) + 2, &size) < 0
123         || NetpULONGPtrRoundUp(size, 2, &size) < 0
124         || RtlULONGAdd(size, *((_WORD *)v3 + 130) + 2, &size) < 0
125         || NetpULONGPtrRoundUp(size, 2, &size) < 0
126         || RtlULONGAdd(size, *((_WORD *)v3 + 134) + 2, &size) < 0
127         || NetpULONGPtrRoundUp(size, 2, &size) < 0
128         || RtlULONGAdd(size, *((_WORD *)v3 + 138) + 2, &size) < 0
129         || NetpULONGPtrRoundUp(size, 2, &size) < 0
130         || RtlULONGAdd(size, *((_WORD *)v3 + 142) + 2, &size) < 0
131         || NetpULONGPtrRoundUp(size, 2, &size) < 0
132         || RtlULONGAdd(size, *((_WORD *)v3 + 146) + 2, &size) < 0 )
133     {
134         NlPrintRoutine(256, L"NlpAddResourceGroupsToSamInfo: Integer overflow in length calculation at line %d\n",
135             return -1073741675;
136     }
137 }
```

Windows 7

Integer overflow

```
.text:06D161E0 ; Attributes: bp-based frame
.text:06D161E0
.text:06D161E0 ; int __stdcall _EscapeField(LPCWSTR psz, int)
.text:06D161E0 ?_EscapeField@@YGJBPAPAG@Z proc near ; CODE XREF: SHGetParsingNameFromProperty
.text:06D161E0
.text:06D161E0 var_C          = dword ptr -0Ch
.text:06D161E0 uBytes        = dword ptr -8
.text:06D161E0 var_4         = dword ptr -4
.text:06D161E0 psz          = dword ptr 8
.text:06D161E0 arg_4        = dword ptr 0Ch
.text:06D161E0
.text:06D161E0          mov     edi, edi
.text:06D161E2          push  ebp
.text:06D161E3          mov     ebp, esp
.text:06D161E5          mov     eax, [ebp+arg_4]
.text:06D161E8          and     dword ptr [eax], 0
.text:06D161EB          sub     esp, 0Ch
.text:06D161EE          push  edi
.text:06D161EF          push  [ebp+psz]          ; lpString
.text:06D161F2          call   ds:__imp__lstrlenW@@4 ; lstrlenW(x)
.text:06D161F8          mov     edi, eax
.text:06D161FA          imul   edi, 3
.text:06D161FD          inc     edi
.text:06D161FE          cmp     edi, 20000h
.text:06D16204          ja     loc_6D16307
.text:06D16208          and     [ebp+uBytes], 0
.text:06D1620E          lea   eax, [ebp+uBytes]
.text:06D16211          push  eax                ; uBytes
.text:06D16212          push  edi                ; int
.text:06D16213          call   ??$LocalAllocArray@@@YGJIPAPAG@Z ; LocalAllocArray<ushor
.text:06D16218          mov     [ebp+var_4], eax
```


Windows 8 /ULongLongToUint Patched!

```
0 ; Attributes: bp-based frame
1
2 ; int __stdcall _EscapeField(LPCWSTR psz, int)
3 ?_EscapeField@@YGJPBGAPAG@Z proc near ; CODE XREF: SHGetParsingNameFromPropertyS
4
5 var_C          = dword ptr -0Ch |
6 var_8          = dword ptr -8
7 uBytes         = dword ptr -4
8 psz           = dword ptr 8
9 arg_4         = dword ptr 0Ch
10
11 mov     edi, edi
12 push  ebp
13 mov     ebp, esp
14 mov     eax, [ebp+arg_4]
15 sub     esp, 0Ch
16 push  ebx
17 mov     ebx, [ebp+psz]
18 push  esi
19 xor     esi, esi
20 push  edi
21 mov     [eax], esi
22 lea    eax, [ebp+var_8]
23 push  eax ; unsigned int *
24 push  ebx ; lpString
25 call  _lstrlenW@4 ; lstrlenW(x)
26 push  3
27 pop   ecx
28 mul   ecx
29 push  edx
30 push  eax ; unsigned __int64
31 call  ?ULongLongToUInt@@YGJ_KPAI@Z ; ULongLongToUInt(unsigned __
32 test  eax, eax
33 jmp  77014500
```

Whats Next

- Possible Extensions
 - Win8, we're coming!!
 - Extended signatures
- Improvements
 - Transparent DB library
 - Integration of components
- Known issues
 - Duplicate hits, false positives, slooow, output is not handy

Happy Diffing.



"Most of all I love your vulnerability."