

A stack of papers is shown against a dark blue background. The top paper is light blue and contains several lines of binary code (0s and 1s). Below it, a white paper is partially visible, showing assembly code snippets. The text 'Hopper Disassembler' is overlaid in large white font across the center of the papers.

Hopper Disassembler

Cryptic Apps - Vincent Bénony (bsr43@hopperapp.com)

Qu'est-ce que Hopper ?

- ▶ un désassembleur interactif
 - ▶ Intel x86, et ARM
 - ▶ Mach-O, Windows PE, ELF
- ▶ un décompilateur
- ▶ un debugger

L'interface

FlashCode.hop

Read Executable Back Follow Mark As Data Mark As ASCII Mark As Code Mark As Procedure Segments Cross References Show CFG Pseudo Code

Labels Strings

Search

EntryPoint
sub_10001dfc
methImpl_AppDelegate_registrationPublicKey
methImpl_AppDelegate_checkRegistrationL...
methImpl_AppDelegate_registrationName...
methImpl_AppDelegate_registrationOrderID...
methImpl_AppDelegate_registrationData...
methImpl_AppDelegate_registrationName...
methImpl_AppDelegate_registrationOrderID...
methImpl_AppDelegate_checkSavedRegistration...
methImpl_AppDelegate_installLicenseData...
methImpl_AppDelegate_installLicenseAdPat...
methImpl_AppDelegate_installLicenseAdPat...
methImpl_AppDelegate_feedParametersFor...
methImpl_AppDelegate_registerObserverO...
methImpl_AppDelegate_unregisterObserver...
methImpl_AppDelegate_characterOfEntry...
methImpl_AppDelegate_entryFromCharacter...
methImpl_AppDelegate_observeValueForKe...
methImpl_AppDelegate_dealloc
methImpl_AppDelegate_switchToMainCont...
methImpl_AppDelegate_init
methImpl_AppDelegate_applicationWillFinis...
methImpl_AppDelegate_updater
methImpl_AppDelegate_applicationDidFinis...
methImpl_AppDelegate_launchUpdate...
methImpl_AppDelegate_setClassPrefix...
methImpl_AppDelegate_classPrefix
methImpl_AppDelegate_play...
methImpl_AppDelegate_setTargetIndex...
methImpl_AppDelegate_targetIndex

```
00000001000027f4 415E      pop     r14
00000001000027f6 415F      pop     r15
00000001000027f8 5D       pop     rbp
00000001000027f9 C3       ret

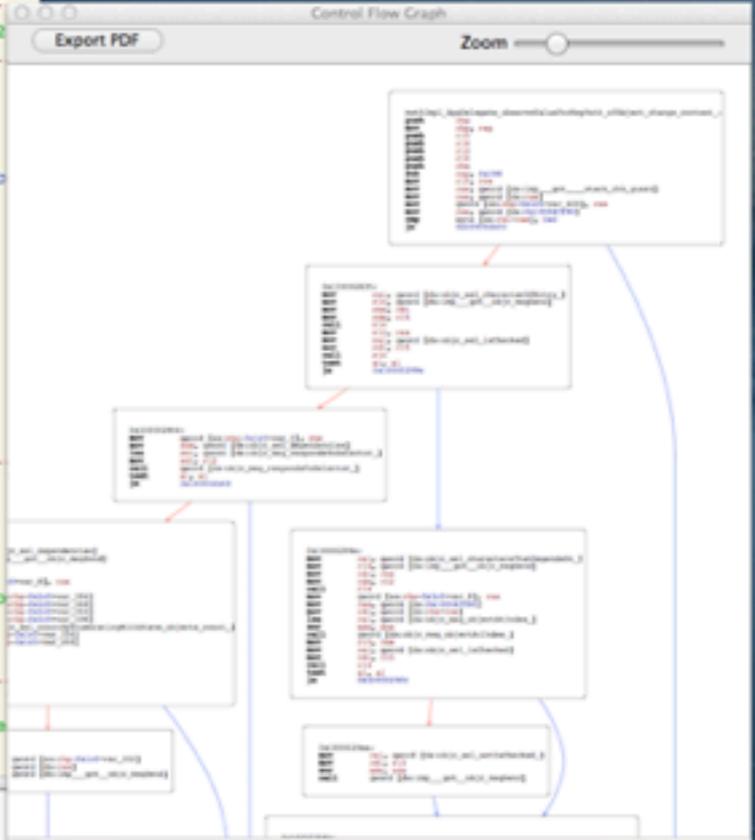
; Basic Block Input Regs: <nothing> - Killed Regs: <nothing>
00000001000027fa E82D9E0200 call    imp_stub_stack_chk_fail ; XREF=0x1000027fa
; endp

----- BEGIN OF PROCEDURE -----

; Basic Block Input Regs: rcx rdi - Killed Regs: rax rbp r15
methImpl_AppDelegate_observeValueForKeyPath_ofObject_change_c
00000001000027ff 55       push   rbp
0000000100002800 4889E5   mov    rbp, rbp
0000000100002803 4157     push  r15
0000000100002805 4156     push  r14
0000000100002807 4155     push  r13
0000000100002809 4154     push  r12
000000010000280b 53       push  rbx
000000010000280c 4881EC98010000 sub    rsp, 0x198
000000010000280e 4989CF   mov    r15, rcx
0000000100002810 488B0543F80300 mov    rax, qword [ds:imp__got_stack_chk_guard]
0000000100002811 488B00   mov    rax, qword [ds:rax]
0000000100002812 488945D0 mov    qword [ss:rbp-0x1c0+var_400], rax
0000000100002814 488B0565D70400 mov    rax, qword [ds:0x10004ff90]
0000000100002816 803C0700 cmp    byte [ds:rdi+rax], 0x0
0000000100002818 7E       je     0x100002b03

; Basic Block Input Regs: rax rdi r15 - Killed Regs: rdx rbx rsi
0000000100002835 488B35FC7B0400 mov    rsi, qword [ds:objc_sel_characterOfEntry_] ;
0000000100002836 4C8B353DF80300 mov    r14, qword [ds:imp__got_objc_msgSend]
0000000100002838 48897B   mov    rbx, rdi
0000000100002839 4C897A   mov    rdx, r15
000000010000283b 417FD6   call   r14
000000010000283d 4989C4   mov    r12, rax
000000010000283f 488B35EA7B0400 mov    rsi, qword [ds:objc_sel_isChecked] ; #selecto
0000000100002840 4C89FF   mov    rdi, r15
0000000100002842 417FD6   call   r14
0000000100002844 84C0    test   al, al
0000000100002846 7E       je     0x10000299e

; Basic Block Input Regs: rax rbx r12 - Killed Regs: rdx rbp rsi
0000000100002864 48899D40FEFFFF mov    qword [ss:rbp-0x1c0+var_0], rbx
0000000100002866 488B15D67B0400 mov    rdx, qword [ds:objc_sel_dependencies] ; #sele
0000000100002868 488D3597870400 lea   rsi, qword [ds:objc_msg_respondsToSelector_]
0000000100002869 4C89E7   mov    rdi, r12
```



Pseudo Code

```
function methImpl_AppDelegate_observeValueForKeyPath_ofObject_change_context_ {
    r15 = rcx;
    var_400 = **__stack_chk_guard;
    if (*(int0_t*)(rdi + *0x10004ff90) != 0x0) {
        r14 = *objc_msgSend;
        rbx = rdi;
        rax = [rdi characterOfEntry:r15];
        r12 = rax;
        rax = [r15 isChecked];
        if (rax != 0x0) {
            var_0 = rbx;
            rax = [r12 respondsToSelector:@selector(dependencies)];
            if (rax != 0x0) {
                r14 = *objc_msgSend;
                rax = [r12 dependencies];
                var_8 = rax;
                rax = [rax countByEnumeratingWithState:&var_336 objects:&var_208 count:0x10];
                if (rax != 0x0) {
                    r15 = *var_352;
                    do {
                        r13 = 0x0;
                        do {
                            if (*var_352 != r15) {
                                objc_enumerationMutation(var_8);
                            }
                        } while (r13 != 0x0);
                    } while (r15 != 0x0);
                }
            }
        }
    }
}
```

Crypto Tools.hop

Read Executable Back Follow

Mark As Data Mark As ASCII Mark As Code Mark As Procedure

Symbol Names Segments Cross References Show CFG Pseudo Code

GDB

Label Name

EntryPoint

0000000010000191a 90 nop

0000000010000191b 90 nop

0000000010000191c 90 nop

0000000010000191d 90 nop

0000000010000191e 90 nop

0000000010000191f 90 nop

***** BEGIN OF PROCEDURE *****

; Basic Block Input Regs: rax rsi rdi - Killed Regs: rbp rsi rdi

methImpl_Crypto_ToolsAppDelegate_setHashControlEnableState:

00000000100001920 55 push rbp

00000000100001921 4889E5 mov rbp, rbp

00000000100001922 4883EC50 sub rsp, 0x50

00000000100001928 48897DF8 mov qword [ss:rbp-0x50+var_72], rdi

0000000010000192c 488975F0 mov qword [ss:rbp-0x50+var_64], rsi

00000000100001930 488B75F8 mov rsi, qword [ss:rbp-0x50+var_72]

00000000100001934 488B3D354B0300 mov rdi, qword [ds:OBJC_IVAR_\$_Crypto_ToolsAppDelegate.hashSourceMat

0000000010000193b 488B343E mov rsi, qword [ds:rsi+rdi]

0000000010000193f 488975E8 mov qword [ss:rbp-0x50+var_56], rsi

00000000100001943 488B3D6620300 mov rsi, qword [ds:objc_sel_selectedRow] ; #selector(selectedRow)

0000000010000194a 488B7D88 mov rdi, qword [ss:rbp-0x50+var_54]

0000000010000194e E88DCD0200 call imp_stubs_objc_msgSend

00000000100001953 483D00000000 cmp rax, 0x0

00000000100001959 0F8577F0000000 jne 0x1000019de

; Basic Block Input Regs: rbp - Killed Regs: rax rcx rdx rbp rsi rdi

0000000010000195f B000 mov al, 0x0

00000000100001961 B101 mov cl, 0x1

00000000100001963 488B55F8 mov rdx, qword [ss:rbp-0x50+var_72]

00000000100001967 488B350A4B0300 mov rsi, qword [ds:OBJC_IVAR_\$_Crypto_ToolsAppDelegate.hashSourceStr

0000000010000196e 488B1432 mov rdx, qword [ds:rdx+rsi]

00000000100001972 488B35AF620300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled:)

00000000100001979 4889D7 mov rdi, rdx

0000000010000197c 0FBED1 movsx edx, cl

0000000010000197f 8A45E7 mov byte [ss:rbp-0x50+var_55], al

00000000100001982 E859CD0200 call imp_stubs_objc_msgSend

00000000100001987 488B75F8 mov rsi, qword [ss:rbp-0x50+var_72]

0000000010000198b 488B3DDE4A0300 mov rdi, qword [ds:OBJC_IVAR_\$_Crypto_ToolsAppDelegate.hashSourceFil

00000000100001992 488B343E mov rsi, qword [ds:rsi+rdi]

00000000100001996 488975D8 mov qword [ss:rbp-0x50+var_40], rsi

0000000010000199a 488B3587620300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled:)

000000001000019a1 488B7D08 mov rdi, qword [ss:rbp-0x50+var_40]

000000001000019a5 8A45E7 mov al, byte [ss:rbp-0x50+var_55]

000000001000019a8 0FBED0 movsx edx, al

000000001000019ab E839CD0200 call imp_stubs_objc_msgSend

000000001000019b0 488B75F8 mov rsi, qword [ss:rbp-0x50+var_72]

000000001000019b4 488B3DCC4A0300 mov rdi, qword [ds:OBJC_IVAR_\$_Crypto_ToolsAppDelegate.hashBrowseBut

000000001000019bb 488B343E mov rsi, qword [ds:rsi+rdi]

000000001000019bf 488975D0 mov qword [ss:rbp-0x50+var_32], rsi

Analysis segment: __text

Analysis segment: __objc_classlist

Analysis segment: __objc_protolist

Analysis segment: __objc_imageinfo

Analysis segment: __objc_const

Analysis segment: __objc_selrefs

Analysis segment: __objc_msgrefs

Analysis segment: __objc_classrefs

Analysis segment: __objc_superrefs

Analysis segment: __objc_data

Analysis segment: __data

Analysis segment: __cstring

Analysis segment: __bss

Background analysis ended

Address 0x10000193b, Segment __text, methImpl_Crypto_ToolsAppDelegate_setHashControlEnableState + 27, file offset 0x193b

0000000010000193b 90 nop

0000000010000193c 90 nop

0000000010000193d 90 nop

0000000010000193e 90 nop

0000000010000193f 90 nop

Crypto Tools.hop

D A C P N S X G

Read Executable Back Follow Mark As Data Mark As ASCII Mark As Code Mark As Procedure Symbol Names Segments Cross References Show CFC Pseudo Code CDB

Label Name

```

000000010000191a 90 nop
000000010000191b 90 nop
000000010000191c 90 nop
000000010000191d 90 nop
000000010000191e 90 nop
000000010000191f 90 nop
-----
; BEGIN OF PROCEDURE -----
; Basic Block Input Regs: rax rsi rdi - Killed Regs: rbp rsi rdi
methImpl_Crypto_ToolsAppDelegate_setHashControlEnabledState:
0000000100001920 55 push rbp
0000000100001921 4889E5 mov rbp, rbp
0000000100001924 4883E5 sub rbp, rbp
0000000100001928 48897CFF mov qword [rsi:rbp-0x50+var_72], rdi
000000010000192c 488975FF mov qword [rsi:rbp-0x50+var_64], rsi
0000000100001930 488B75FF mov rsi, qword [rsi:rbp-0x50+var_72]
0000000100001934 488B3D354A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceMat]
000000010000193b 488B343E mov rdi, qword [ds:rax+rdi]
000000010000193f 488B7588 mov qword [rsi:rbp-0x50+var_56], rsi
0000000100001943 488B3D354A0300 mov rsi, qword [ds:objc_sel_selectedShow] ; #selector(selectedShow)
000000010000194a 488B7D88 mov rdi, qword [rsi:rbp-0x50+var_56]
000000010000194e 488B3D354A0300 call imp__stabs_objc_msgSend
0000000100001953 483D00000000 cmp rax, 0
0000000100001959 48757F000000 jne 0x10000195e
; Basic Block Input Regs: rbp - Killed Regs: rax rcx rdx rbp rsi rdi
000000010000195f 3000 mov al, 0x0
0000000100001961 3101 cl, 0x1
0000000100001963 488B55FF mov rdx, qword [rsi:rbp-0x50+var_72]
0000000100001967 488B353A4A0300 mov rsi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceStr]
000000010000196e 488B3432 mov rdx, qword [ds:rax+rdi]
0000000100001972 488B353A4A0300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
0000000100001979 488B7D7F mov rdi, rdx
000000010000197c 0FBE01 movzx edx, cl
000000010000197f 8B45E7 mov byte [rsi:rbp-0x50+var_55], al
0000000100001982 488B3D354A0300 call imp__stabs_objc_msgSend
0000000100001987 488B75FF mov rsi, qword [rsi:rbp-0x50+var_72]
000000010000198b 488B3D354A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceFile]
0000000100001992 488B343E mov rsi, qword [ds:rax+rdi]
0000000100001996 488B7508 mov qword [rsi:rbp-0x50+var_40], rsi
000000010000199a 488B353A4A0300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
00000001000019a1 488B7D08 mov rdi, qword [rsi:rbp-0x50+var_40]
00000001000019a5 8B45E7 mov al, byte [rsi:rbp-0x50+var_55]
00000001000019a8 0FBE01 movzx edx, al
00000001000019ab 488B3D354A0300 call imp__stabs_objc_msgSend
00000001000019b0 488B75FF mov rsi, qword [rsi:rbp-0x50+var_72]
00000001000019b4 488B3D354A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashBrowseBut]
00000001000019bb 488B343E mov rsi, qword [ds:rax+rdi]
00000001000019bf 488B7500 mov qword [rsi:rbp-0x50+var_32], rsi

```

Analysis segment: __ic_string_pur
Analysis segment: __objc_classref
Analysis segment: __objc_protocol
Analysis segment: __objc_imageinfo
Analysis segment: __objc_const
Analysis segment: __objc_selref
Analysis segment: __objc_mangler
Analysis segment: __objc_classref
Analysis segment: __objc_superref
Analysis segment: __objc_data
Analysis segment: __data
Analysis segment: __cstring
Analysis segment: __bss
Background analysis ended

Address 0x10000193b, Segment __text, methImpl_Crypto_ToolsAppDelegate_setHashControlEnabledState + 27, File offset 0x193b

```

000000010000193b 488B343E mov rsi, qword [ds:rax+rdi]
000000010000193f 488B7588 mov qword [rsi:rbp-0x50+var_56], rsi
0000000100001943 488B3D354A0300 mov rsi, qword [ds:objc_sel_selectedShow] ; #selector(selectedShow)
000000010000194a 488B7D88 mov rdi, qword [rsi:rbp-0x50+var_56]
000000010000194e 488B3D354A0300 call imp__stabs_objc_msgSend
0000000100001953 483D00000000 cmp rax, 0
0000000100001959 48757F000000 jne 0x10000195e
; Basic Block Input Regs: rbp - Killed Regs: rax rcx rdx rbp rsi rdi
000000010000195f 3000 mov al, 0x0
0000000100001961 3101 cl, 0x1
0000000100001963 488B55FF mov rdx, qword [rsi:rbp-0x50+var_72]
0000000100001967 488B353A4A0300 mov rsi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceStr]
000000010000196e 488B3432 mov rdx, qword [ds:rax+rdi]
0000000100001972 488B353A4A0300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
0000000100001979 488B7D7F mov rdi, rdx
000000010000197c 0FBE01 movzx edx, cl
000000010000197f 8B45E7 mov byte [rsi:rbp-0x50+var_55], al
0000000100001982 488B3D354A0300 call imp__stabs_objc_msgSend
0000000100001987 488B75FF mov rsi, qword [rsi:rbp-0x50+var_72]
000000010000198b 488B3D354A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceFile]
0000000100001992 488B343E mov rsi, qword [ds:rax+rdi]
0000000100001996 488B7508 mov qword [rsi:rbp-0x50+var_40], rsi
000000010000199a 488B353A4A0300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
00000001000019a1 488B7D08 mov rdi, qword [rsi:rbp-0x50+var_40]
00000001000019a5 8B45E7 mov al, byte [rsi:rbp-0x50+var_55]
00000001000019a8 0FBE01 movzx edx, al
00000001000019ab 488B3D354A0300 call imp__stabs_objc_msgSend
00000001000019b0 488B75FF mov rsi, qword [rsi:rbp-0x50+var_72]
00000001000019b4 488B3D354A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashBrowseBut]
00000001000019bb 488B343E mov rsi, qword [ds:rax+rdi]
00000001000019bf 488B7500 mov qword [rsi:rbp-0x50+var_32], rsi

```

```
0000000010000191a 90 nop
0000000010000191b 90 nop
0000000010000191c 90 nop
0000000010000191d 90 nop
0000000010000191e 90 nop
0000000010000191f 90 nop

----- BEGIN OF PROCEDURE -----
; Basic Block Input Regs: rax rsi rdi - Killed Regs: rbp rsi rdi
methImpl_Crypto_ToolsAppDelegate_setHashControlEnabledState:
00000000100001920 55 push rbp
00000000100001921 4889E5 mov rbp, rbp
00000000100001924 4883EC50 sub rsp, 0x50
00000000100001928 48897C9F mov qword [rsi:rbp-0x50+var_72], rdi
0000000010000192c 488975F0 mov qword [rsi:rbp-0x50+var_64], rsi
00000000100001930 488975F8 mov rsi, qword [rsi:rbp-0x50+var_72]
00000000100001934 488B3035480300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceMat]
0000000010000193b 488B343E mov rdi, qword [ds:rxi+rdi]
0000000010000193f 488973E8 mov qword [rsi:rbp-0x50+var_56], rsi
00000000100001943 488B3204620300 mov rxi, qword [ds:objc_sel_selectedShow] ; #selector(selectedShow)
0000000010000194a 48897C08 mov rdi, qword [rsi:rbp-0x50+var_56]
0000000010000194e 288DC0200 call imp__stabs_objc_msgSend
00000000100001953 483D00000000 cmp rax, 0x0
00000000100001959 0F957F000000 jne 0x10000195e

; Basic Block Input Regs: rbp - Killed Regs: rax rcx rdx rbp rsi rdi
0000000010000195f 3000 mov al, 0x0
00000000100001961 3101 cl, 0x1
00000000100001963 488B55F8 mov rdx, qword [rsi:rbp-0x50+var_72]
00000000100001967 488B350A480300 mov rxi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceStr]
0000000010000196e 488B3432 mov rdx, qword [ds:rxi+rxi]
00000000100001972 488B35AF620300 mov rxi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
00000000100001979 4889D7 mov rdi, rdx
0000000010000197c 0F8ED1 movzx edx, cl
0000000010000197f 8B45E7 mov byte [rsi:rbp-0x50+var_55], al
00000000100001982 2859C0200 call imp__stabs_objc_msgSend
00000000100001987 488975F8 mov rxi, qword [rsi:rbp-0x50+var_72]
0000000010000198b 488B30EE4A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceFil]
00000000100001992 488B343E mov rxi, qword [ds:rxi+rdi]
00000000100001996 48897508 mov qword [rsi:rbp-0x50+var_40], rxi
0000000010000199a 488B3587620300 mov rxi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
000000001000019a1 488B7C08 mov rdi, qword [rsi:rbp-0x50+var_40]
000000001000019a5 8B45E7 mov al, byte [rsi:rbp-0x50+var_55]
000000001000019a8 0F8ED0 movzx edx, al
000000001000019ab 283DC0200 call imp__stabs_objc_msgSend
000000001000019b0 488975F8 mov rxi, qword [rsi:rbp-0x50+var_72]
000000001000019b4 488B30CD4A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashBrowseBut]
000000001000019bb 488B343E mov rxi, qword [ds:rxi+rdi]
000000001000019bf 48897500 mov qword [rsi:rbp-0x50+var_32], rxi
```

Code désassemblé

```
0000000010000191a 90 nop
0000000010000191b 90 nop
0000000010000191c 90 nop
0000000010000191d 90 nop
0000000010000191e 90 nop
0000000010000191f 90 nop
```

```
0000010000191a 90 nop
0000010000191b 90 nop
0000010000191c 90 nop
0000010000191d 90 nop
0000010000191e 90 nop
0000010000191f 90 nop

-----BEGIN OF PROCEDURE-----

; Basic Block Input Regs: rax rsi rdi - Killed Regs: rbp rsi rdi
methimpl_Crypto_ToolsAppDelegate_setHashControlEnabledState:
00000100001920 55 push rbp
00000100001921 4889E5 mov rbp, rbp
00000100001922 4883EC50 sub rsp, 0x50
00000100001923 48897C9F mov qword [rsi:rbp-0x50+var_72], rdi
00000100001924 488975F0 mov qword [rsi:rbp-0x50+var_64], rsi
00000100001925 488B75F8 mov rsi, qword [rsi:rbp-0x50+var_72]
00000100001926 488B3D30 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceMat]
00000100001927 488B343E mov rdi, qword [ds:rsi+rdi]
00000100001928 488B73E8 mov qword [rsi:rbp-0x50+var_56], rsi
00000100001929 488B3D420300 mov rsi, qword [ds:objc_sel_selectedShow] ; #selector(selectedShow)
0000010000192a 488B7C08 mov rdi, qword [rsi:rbp-0x50+var_56]
0000010000192b 488BDC0200 call imp__stabs_objc_msgSend
0000010000192c 488B3D0000 mov rax, 0x0
0000010000192d 488B3D0000 jmp 0x10000192e

; Basic Block Input Regs: rbp - Killed Regs: rax rcx rdx rbp rsi rdi
0000010000192e 3000 mov al, 0x0
0000010000192f 3101 cl, 0x1
00000100001930 488B55F8 mov rdx, qword [rsi:rbp-0x50+var_72]
00000100001931 488B35A40300 mov rsi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceStr]
00000100001932 488B4322 mov rdx, qword [ds:rdx+rsi]
00000100001933 488B35AF620300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
00000100001934 4889D7 mov rdi, rdx
00000100001935 0FBE01 movzx edx, cl
00000100001936 8B45E7 mov byte [rsi:rbp-0x50+var_55], al
00000100001937 488BDC0200 call imp__stabs_objc_msgSend
00000100001938 488B75F8 mov rsi, qword [rsi:rbp-0x50+var_72]
00000100001939 488B3D0E4A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceFil]
0000010000193a 488B343E mov rsi, qword [ds:rsi+rdi]
0000010000193b 48897508 mov qword [rsi:rbp-0x50+var_40], rsi
0000010000193c 488B3587620300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
0000010000193d 488B7C08 mov rdi, qword [rsi:rbp-0x50+var_40]
0000010000193e 0FBE01 movzx edx, al
0000010000193f 488BDC0200 call imp__stabs_objc_msgSend
00000100001940 488B75F8 mov rsi, qword [rsi:rbp-0x50+var_72]
00000100001941 488B3D0E4A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashBrowseBut]
00000100001942 488B343E mov rsi, qword [ds:rsi+rdi]
00000100001943 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
```

Signets
et
Points d'arrêt

```
0000010000191a 90 nop
0000010000191b 90 nop
0000010000191c 90 nop
0000010000191d 90 nop
0000010000191e 90 nop
0000010000191f 90 nop
```

The image shows a debugger window titled "Crypto Tools.hop". The assembly view displays the following code:

```
000000001000191a 90 nop
000000001000191b 90 nop
000000001000191c 90 nop
000000001000191d 90 nop
000000001000191e 90 nop
000000001000191f 90 nop

----- BEGIN OF PROCEDURE -----

; Basic Block Input Regs: rax rsi rdi - Killed Regs: rbp rsi rdi
methmpl_Crypto_ToolsAppDelegate_setHashControlEnabledState:
push rbp
mov rbp, rbp
sub rsp, 0x50
mov qword [rsi:rbp-0x50+var_72], rdi
mov qword [rsi:rbp-0x50+var_64], rsi
mov rsi, qword [rsi:rbp-0x50+var_72]
mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceMat]
mov rdx, qword [ds:rax+rdi]
mov qword [rsi:rbp-0x50+var_56], rdx
mov rsi, qword [ds:objc_sel_selectedShow] ; #selector(selectedShow)
mov rdi, qword [rsi:rbp-0x50+var_56]
call imp__stabs_objc_msgSend
mov rax, 0x0
cmp rax, 0x0
jne 0x1000019de

; Basic Block Input Regs: rbp - Killed Regs: rax rcx rdx rbp rsi rdi
mov al, 0x0
mov cl, 0x1
mov rdx, qword [rsi:rbp-0x50+var_72]
mov rsi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceStr]
mov rdx, qword [ds:rax+rdi]
mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
mov rdi, rdx
movzx edx, cl
mov byte [rsi:rbp-0x50+var_55], al
call imp__stabs_objc_msgSend
mov rsi, qword [rsi:rbp-0x50+var_72]
mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceFil]
mov rsi, qword [ds:rax+rdi]
mov qword [rsi:rbp-0x50+var_40], rsi
mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
mov rdi, qword [rsi:rbp-0x50+var_40]
movzx al, byte [rsi:rbp-0x50+var_55]
movzx edx, al
call imp__stabs_objc_msgSend
mov rsi, qword [rsi:rbp-0x50+var_72]
mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashBrowseBut]
mov rsi, qword [ds:rax+rdi]
mov qword [rsi:rbp-0x50+var_32], rsi
```

Flot d'exécution
du programme

```
0000000010000222 90 nop
0000000010000223 90 nop
0000000010000224 90 nop
0000000010000225 90 nop
0000000010000226 90 nop
0000000010000227 90 nop
0000000010000228 90 nop
0000000010000229 90 nop
000000001000022a 90 nop
000000001000022b 90 nop
000000001000022c 90 nop
000000001000022d 90 nop
000000001000022e 90 nop
000000001000022f 90 nop
0000000010000230 90 nop
0000000010000231 90 nop
0000000010000232 90 nop
0000000010000233 90 nop
0000000010000234 90 nop
0000000010000235 90 nop
0000000010000236 90 nop
0000000010000237 90 nop
0000000010000238 90 nop
0000000010000239 90 nop
000000001000023a 90 nop
000000001000023b 90 nop
000000001000023c 90 nop
000000001000023d 90 nop
000000001000023e 90 nop
000000001000023f 90 nop
0000000010000240 90 nop
0000000010000241 90 nop
0000000010000242 90 nop
0000000010000243 90 nop
0000000010000244 90 nop
0000000010000245 90 nop
0000000010000246 90 nop
0000000010000247 90 nop
0000000010000248 90 nop
0000000010000249 90 nop
000000001000024a 90 nop
000000001000024b 90 nop
000000001000024c 90 nop
000000001000024d 90 nop
000000001000024e 90 nop
000000001000024f 90 nop
0000000010000250 90 nop
0000000010000251 90 nop
0000000010000252 90 nop
0000000010000253 90 nop
0000000010000254 90 nop
0000000010000255 90 nop
0000000010000256 90 nop
0000000010000257 90 nop
0000000010000258 90 nop
0000000010000259 90 nop
000000001000025a 90 nop
000000001000025b 90 nop
000000001000025c 90 nop
000000001000025d 90 nop
000000001000025e 90 nop
000000001000025f 90 nop
0000000010000260 90 nop
0000000010000261 90 nop
0000000010000262 90 nop
0000000010000263 90 nop
0000000010000264 90 nop
0000000010000265 90 nop
0000000010000266 90 nop
0000000010000267 90 nop
0000000010000268 90 nop
0000000010000269 90 nop
000000001000026a 90 nop
000000001000026b 90 nop
000000001000026c 90 nop
000000001000026d 90 nop
000000001000026e 90 nop
000000001000026f 90 nop
0000000010000270 90 nop
0000000010000271 90 nop
0000000010000272 90 nop
0000000010000273 90 nop
0000000010000274 90 nop
0000000010000275 90 nop
0000000010000276 90 nop
0000000010000277 90 nop
0000000010000278 90 nop
0000000010000279 90 nop
000000001000027a 90 nop
000000001000027b 90 nop
000000001000027c 90 nop
000000001000027d 90 nop
000000001000027e 90 nop
000000001000027f 90 nop
0000000010000280 90 nop
0000000010000281 90 nop
0000000010000282 90 nop
0000000010000283 90 nop
0000000010000284 90 nop
0000000010000285 90 nop
0000000010000286 90 nop
0000000010000287 90 nop
0000000010000288 90 nop
0000000010000289 90 nop
000000001000028a 90 nop
000000001000028b 90 nop
000000001000028c 90 nop
000000001000028d 90 nop
000000001000028e 90 nop
000000001000028f 90 nop
0000000010000290 90 nop
0000000010000291 90 nop
0000000010000292 90 nop
0000000010000293 90 nop
0000000010000294 90 nop
0000000010000295 90 nop
0000000010000296 90 nop
0000000010000297 90 nop
0000000010000298 90 nop
0000000010000299 90 nop
000000001000029a 90 nop
000000001000029b 90 nop
000000001000029c 90 nop
000000001000029d 90 nop
000000001000029e 90 nop
000000001000029f 90 nop
00000000100002a0 90 nop
00000000100002a1 90 nop
00000000100002a2 90 nop
00000000100002a3 90 nop
00000000100002a4 90 nop
00000000100002a5 90 nop
00000000100002a6 90 nop
00000000100002a7 90 nop
00000000100002a8 90 nop
00000000100002a9 90 nop
00000000100002aa 90 nop
00000000100002ab 90 nop
00000000100002ac 90 nop
00000000100002ad 90 nop
00000000100002ae 90 nop
00000000100002af 90 nop
00000000100002b0 90 nop
00000000100002b1 90 nop
00000000100002b2 90 nop
00000000100002b3 90 nop
00000000100002b4 90 nop
00000000100002b5 90 nop
00000000100002b6 90 nop
00000000100002b7 90 nop
00000000100002b8 90 nop
00000000100002b9 90 nop
00000000100002ba 90 nop
00000000100002bb 90 nop
00000000100002bc 90 nop
00000000100002bd 90 nop
00000000100002be 90 nop
00000000100002bf 90 nop
00000000100002c0 90 nop
00000000100002c1 90 nop
00000000100002c2 90 nop
00000000100002c3 90 nop
00000000100002c4 90 nop
00000000100002c5 90 nop
00000000100002c6 90 nop
00000000100002c7 90 nop
00000000100002c8 90 nop
00000000100002c9 90 nop
00000000100002ca 90 nop
00000000100002cb 90 nop
00000000100002cc 90 nop
00000000100002cd 90 nop
00000000100002ce 90 nop
00000000100002cf 90 nop
00000000100002d0 90 nop
00000000100002d1 90 nop
00000000100002d2 90 nop
00000000100002d3 90 nop
00000000100002d4 90 nop
00000000100002d5 90 nop
00000000100002d6 90 nop
00000000100002d7 90 nop
00000000100002d8 90 nop
00000000100002d9 90 nop
00000000100002da 90 nop
00000000100002db 90 nop
00000000100002dc 90 nop
00000000100002dd 90 nop
00000000100002de 90 nop
00000000100002df 90 nop
00000000100002e0 90 nop
00000000100002e1 90 nop
00000000100002e2 90 nop
00000000100002e3 90 nop
00000000100002e4 90 nop
00000000100002e5 90 nop
00000000100002e6 90 nop
00000000100002e7 90 nop
00000000100002e8 90 nop
00000000100002e9 90 nop
00000000100002ea 90 nop
00000000100002eb 90 nop
00000000100002ec 90 nop
00000000100002ed 90 nop
00000000100002ee 90 nop
00000000100002ef 90 nop
00000000100002f0 90 nop
00000000100002f1 90 nop
00000000100002f2 90 nop
00000000100002f3 90 nop
00000000100002f4 90 nop
00000000100002f5 90 nop
00000000100002f6 90 nop
00000000100002f7 90 nop
00000000100002f8 90 nop
00000000100002f9 90 nop
00000000100002fa 90 nop
00000000100002fb 90 nop
00000000100002fc 90 nop
00000000100002fd 90 nop
00000000100002fe 90 nop
00000000100002ff 90 nop
0000000010000300 90 nop
```

```
0000000010000191a 90 nop
0000000010000191b 90 nop
0000000010000191c 90 nop
0000000010000191d 90 nop
0000000010000191e 90 nop
0000000010000191f 90 nop

----- BEGIN OF PROCEDURE -----
; Basic Block Input Regs: rax rsi rdi - Killed Regs: rbp rsi rdi
methImpl_Crypto_ToolsAppDelegate_setHashControlEnabledState:
00000000100001920 55 push rbp
00000000100001921 4889E5 mov rbp, rsp
00000000100001922 4883EC50 sub rsp, 0x50
00000000100001923 48897C9F mov qword [rsi:rbp-0x50+var_72], rdi
00000000100001924 488975F0 mov qword [rsi:rbp-0x50+var_64], rsi
00000000100001925 488975F8 mov rsi, qword [rsi:rbp-0x50+var_72]
00000000100001926 488B3035409300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceMat]
00000000100001927 488B343E mov rdi, qword [ds:rsi+rdi]
00000000100001928 488B73E8 mov qword [rsi:rbp-0x50+var_56], rsi
00000000100001929 488B3204620300 mov rsi, qword [ds:objc_sel_selectedShow] ; #selector(selectedShow)
0000000010000192a 488B7C08 mov rdi, qword [rsi:rbp-0x50+var_56]
0000000010000192b 888DC0200 call imp__stabs_objc_msgSend
0000000010000192c 488B3035409300 mov rax, 0x0
0000000010000192d 488B3035409300 jne 0x10000192e

; Basic Block Input Regs: rbp - Killed Regs: rax rcx rdx rbp rsi rdi
0000000010000192e 3000 mov al, 0x0
0000000010000192f 3101 cl, 0x1
00000000100001930 488B55F8 mov rdx, qword [rsi:rbp-0x50+var_72]
00000000100001931 488B353A409300 mov rsi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceStr]
00000000100001932 488B4432 mov rdx, qword [ds:rdx+rsi]
00000000100001933 488B35AF620300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
00000000100001934 4889D7 mov rdi, rdx
00000000100001935 0F8ED1 movzx edx, cl
00000000100001936 8045E7 mov byte [rsi:rbp-0x50+var_55], al
00000000100001937 8859CD0200 call imp__stabs_objc_msgSend
00000000100001938 488B75F8 mov rsi, qword [rsi:rbp-0x50+var_72]
00000000100001939 488B30EE4A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceFil]
0000000010000193a 488B343E mov rsi, qword [ds:rsi+rdi]
0000000010000193b 48897508 mov qword [rsi:rbp-0x50+var_40], rsi
0000000010000193c 488B3587620300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
0000000010000193d 488B7C08 mov rdi, qword [rsi:rbp-0x50+var_40]
0000000010000193e 0F85E7 movzx al, byte [rsi:rbp-0x50+var_55]
0000000010000193f 0F85D0 movzx edx, al
00000000100001940 8859CD0200 call imp__stabs_objc_msgSend
00000000100001941 488B75F8 mov rsi, qword [rsi:rbp-0x50+var_72]
00000000100001942 488B30EE4A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashBrowseBut]
00000000100001943 488B343E mov rsi, qword [ds:rsi+rdi]
00000000100001944 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
```

Liste des labels
et des chaînes
de caractères

```
0000000010000191a 90 nop
0000000010000191b 90 nop
0000000010000191c 90 nop
0000000010000191d 90 nop
0000000010000191e 90 nop
0000000010000191f 90 nop
```

```
000000010000191a 90 nop
000000010000191b 90 nop
000000010000191c 90 nop
000000010000191d 90 nop
000000010000191e 90 nop
000000010000191f 90 nop

----- BEGIN OF PROCEDURE -----

; Basic Block Input Regs: rax rsi rdi - Killed Regs: rbp rsi rdi
methImpl_Crypto_ToolsAppDelegate_setHashControlEnabledState:
0000000100001920 55 push rbp
0000000100001921 4889E5 mov rbp, rbp
0000000100001922 48897C8F sub rsp, 0x50
0000000100001923 48897C8F mov qword [rsi:rbp-0x50+var_72], rdi
0000000100001924 48897C8F mov qword [rsi:rbp-0x50+var_64], rsi
0000000100001925 48897C8F mov rsi, qword [rsi:rbp-0x50+var_72]
0000000100001926 48897C8F mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceMat]
0000000100001927 48897C8F mov rdi, qword [ds:rax+rdi]
0000000100001928 48897C8F mov qword [rsi:rbp-0x50+var_56], rax
0000000100001929 48897C8F mov rsi, qword [ds:objc_sel_selectedShow] ; #selector(selectedShow)
000000010000192a 48897C8F mov rdi, qword [rsi:rbp-0x50+var_56]
000000010000192b 48897C8F call imp__stabs_objc_msgSend
000000010000192c 48897C8F cmp rax, 0x0
000000010000192d 48897C8F jne 0x10000192e

; Basic Block Input Regs: rbp - Killed Regs: rax rcx rdx rbp rsi rdi
000000010000192e 3000 mov al, 0x0
000000010000192f 3101 cl, 0x1
0000000100001930 48897C8F mov rdx, qword [rsi:rbp-0x50+var_72]
0000000100001931 48897C8F mov rsi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceStr]
0000000100001932 48897C8F mov rdx, qword [ds:rax+rsi]
0000000100001933 48897C8F mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
0000000100001934 48897C8F mov rdi, rdx
0000000100001935 48897C8F movzx edx, cl
0000000100001936 48897C8F mov byte [rsi:rbp-0x50+var_55], al
0000000100001937 48897C8F call imp__stabs_objc_msgSend
0000000100001938 48897C8F mov rsi, qword [rsi:rbp-0x50+var_72]
0000000100001939 48897C8F mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceFile]
000000010000193a 48897C8F mov rsi, qword [ds:rax+rdi]
000000010000193b 48897C8F mov qword [rsi:rbp-0x50+var_40], rax
000000010000193c 48897C8F mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
000000010000193d 48897C8F mov rdi, qword [rsi:rbp-0x50+var_40]
000000010000193e 48897C8F movzx edx, al
000000010000193f 48897C8F mov byte [rsi:rbp-0x50+var_55], al
0000000100001940 48897C8F call imp__stabs_objc_msgSend
0000000100001941 48897C8F mov rsi, qword [rsi:rbp-0x50+var_72]
0000000100001942 48897C8F mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashBrowseBut]
0000000100001943 48897C8F mov rsi, qword [ds:rax+rdi]
0000000100001944 48897C8F mov qword [rsi:rbp-0x50+var_32], rax
```

Position dans le fichier et dans le code


```
000000001000191a 90 nop
000000001000191b 90 nop
000000001000191c 90 nop
000000001000191d 90 nop
000000001000191e 90 nop
000000001000191f 90 nop

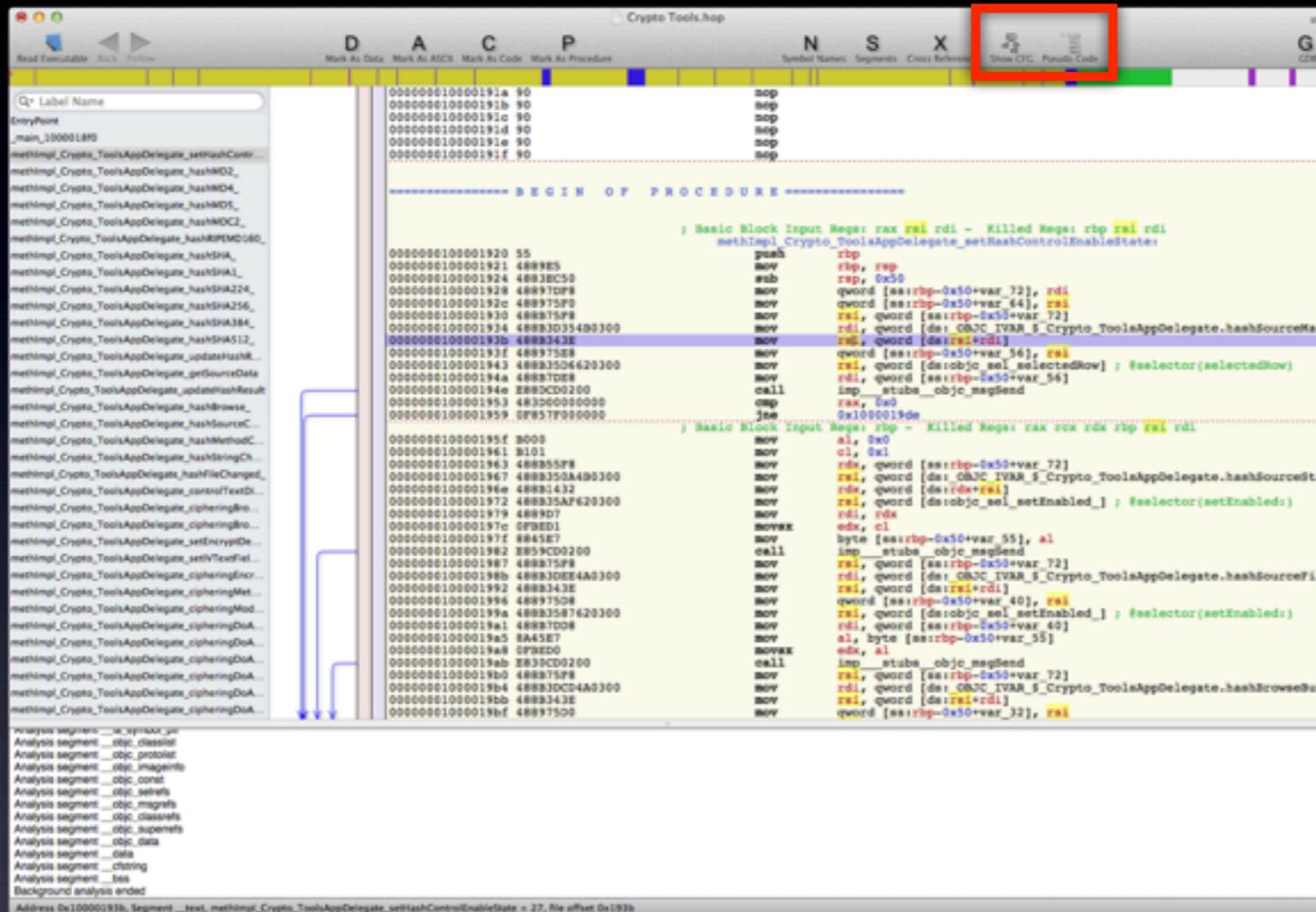
-----BEGIN OF PROCEDURE-----

; Basic Block Input Regs: rax rsi rdi - Killed Regs: rbp rsi rdi
methimpl_Crypto_ToolsAppDelegate_setHashControlEnabledState:
0000000010001920 55 push rbp
0000000010001921 4889E5 mov rbp, rbp
0000000010001922 4883EC50 sub rsp, 0x50
0000000010001923 48897C98 mov qword [rsi:rbp-0x50+var_72], rdi
0000000010001924 488975F0 mov qword [rsi:rbp-0x50+var_64], rsi
0000000010001925 488B75F8 mov rsi, qword [rsi:rbp-0x50+var_72]
0000000010001926 488B3035409300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceMat]
0000000010001927 488B343E mov rdi, qword [ds:rax+rdi]
0000000010001928 488973E8 mov qword [rsi:rbp-0x50+var_56], rsi
0000000010001929 488B3204620300 mov rsi, qword [ds:objc_sel_selectedShow] ; #selector(selectedShow)
000000001000192a 488B7C08 mov rdi, qword [rsi:rbp-0x50+var_56]
000000001000192b 888DC0200 call imp__stabs_objc_msgSend
000000001000192c 488B3035409300 mov rax, 0x0
000000001000192d 488B3035409300 jne 0x10000192e

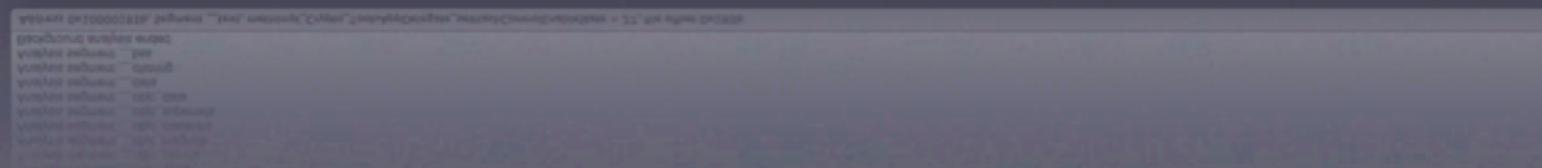
; Basic Block Input Regs: rbp - Killed Regs: rax rcx rdx rbp rsi rdi
000000001000192e 3000 mov al, 0x0
000000001000192f 3101 cl, 0x1
0000000010001930 488B55F8 mov rdx, qword [rsi:rbp-0x50+var_72]
0000000010001931 488B354A409300 mov rsi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceStr]
0000000010001932 488B4432 mov rdx, qword [ds:rax+rdi]
0000000010001933 488B354A409300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
0000000010001934 4889D7 mov rdi, rdx
0000000010001935 0F8ED1 movzx edx, cl
0000000010001936 8845E7 mov byte [rsi:rbp-0x50+var_55], al
0000000010001937 8859CD0200 call imp__stabs_objc_msgSend
0000000010001938 488B75F8 mov rsi, qword [rsi:rbp-0x50+var_72]
0000000010001939 488B30EE4A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashSourceFile]
000000001000193a 488B343E mov rsi, qword [ds:rax+rdi]
000000001000193b 48897508 mov qword [rsi:rbp-0x50+var_40], rsi
000000001000193c 488B3587620300 mov rsi, qword [ds:objc_sel_setEnabled_] ; #selector(setEnabled)
000000001000193d 488B7C08 mov rdi, qword [rsi:rbp-0x50+var_40]
000000001000193e 8845E7 mov al, byte [rsi:rbp-0x50+var_55]
000000001000193f 0F8ED0 movzx edx, al
0000000010001940 8859CD0200 call imp__stabs_objc_msgSend
0000000010001941 488B75F8 mov rsi, qword [rsi:rbp-0x50+var_72]
0000000010001942 488B30EE4A0300 mov rdi, qword [ds:OBJC_IVAR_$_Crypto_ToolsAppDelegate.hashBrowseBut]
0000000010001943 488B343E mov rsi, qword [ds:rax+rdi]
0000000010001944 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
```

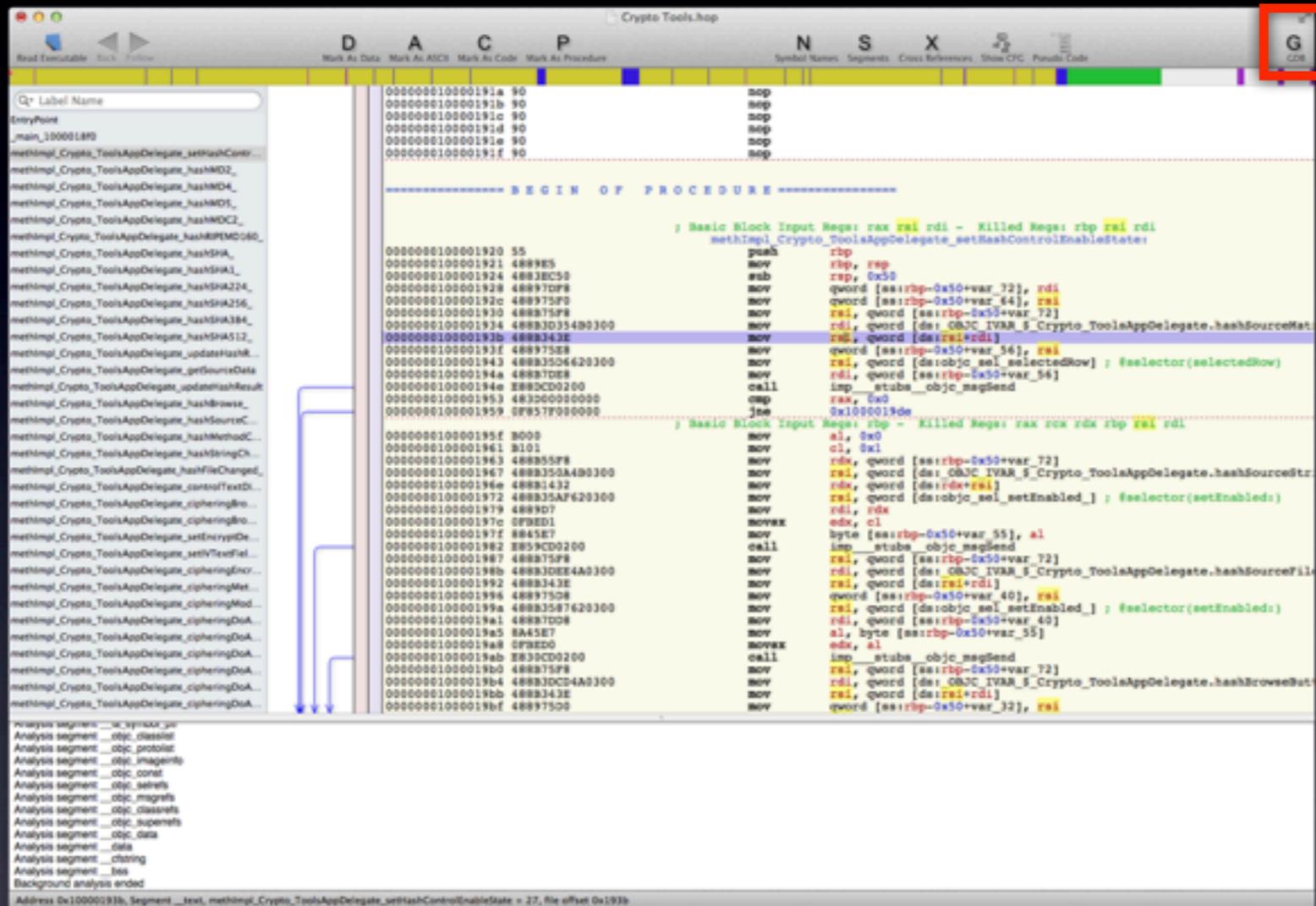
Outils de transformations

```
0000000010000120 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
0000000010000121 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
0000000010000122 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
0000000010000123 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
0000000010000124 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
0000000010000125 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
0000000010000126 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
0000000010000127 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
0000000010000128 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
0000000010000129 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
000000001000012a 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
000000001000012b 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
000000001000012c 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
000000001000012d 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
000000001000012e 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
000000001000012f 48897500 mov qword [rsi:rbp-0x50+var_32], rsi
```

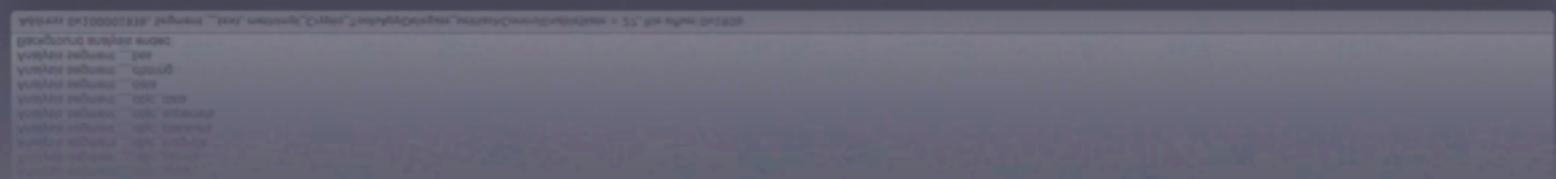



Affichage du graphe de la procédure et du pseudo-code





Debugger



Désassembleur

```
0x55, 0x48, 0x89, 0xe5, 0x48, 0x83, 0xec, 0x10  
0xc7, 0x45, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x89  
0x7d, 0xf8, 0x48, 0x89, 0x75, 0xf0, 0x8b, 0x7d  
0xf8, 0x48, 0x8b, 0x75, 0xf0, 0xe8, 0xcc, 0xcc  
0x02, 0x00, 0x48, 0x83, 0xc4, 0x10, 0x5d, 0xc3  
0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90  
...
```

```

+00  0x55, 0x48, 0x89, 0xe5, 0x48, 0x83, 0xec, 0x10,
+08  0xc7, 0x45, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x89,
+10  0x7d, 0xf8, 0x48, 0x89, 0x75, 0xf0, 0x8b, 0x7d,
+18  0xf8, 0x48, 0x8b, 0x75, 0xf0, 0xe8, 0xcc, 0xcc,
+20  0x02, 0x00, 0x48, 0x83, 0xc4, 0x10, 0x5d, 0xc3,
+28  0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90

```

Adresse	Type
00	Unexplored
01	Unexplored
02	Unexplored
03	Unexplored
04	Unexplored
05	Unexplored
06	Unexplored
07	Unexplored
08	Unexplored
09	Unexplored
0a	Unexplored
⋮	Unexplored

```

+00  0x55,
+01  0x48, 0x89, 0xe5, 0x48, 0x83, 0xec, 0x10,
+08  0xc7, 0x45, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x89,
+10  0x7d, 0xf8, 0x48, 0x89, 0x75, 0xf0, 0x8b, 0x7d,
+18  0xf8, 0x48, 0x8b, 0x75, 0xf0, 0xe8, 0xcc, 0xcc,
+20  0x02, 0x00, 0x48, 0x83, 0xc4, 0x10, 0x5d, 0xc3,
+28  0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90

```

Adresse	Type
00	Code
01	Unexplored
02	Unexplored
03	Unexplored
04	Unexplored
05	Unexplored
06	Unexplored
07	Unexplored
08	Unexplored
09	Unexplored
0a	Unexplored
⋮	Unexplored

```

+00  0x55,
+01  0x48, 0x89, 0xe5,
+04  0x48, 0x83, 0xec, 0x10,
+08  0xc7, 0x45, 0xfc, 0x00, 0x00, 0x00, 0x00, 0x89,
+10  0x7d, 0xf8, 0x48, 0x89, 0x75, 0xf0, 0x8b, 0x7d,
+18  0xf8, 0x48, 0x8b, 0x75, 0xf0, 0xe8, 0xcc, 0xcc,
+20  0x02, 0x00, 0x48, 0x83, 0xc4, 0x10, 0x5d, 0xc3,
+28  0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90

```

Adresse	Type
00	Code
01	Code
02	Next
03	Next
04	Unexplored
05	Unexplored
06	Unexplored
07	Unexplored
08	Unexplored
09	Unexplored
0a	Unexplored
⋮	Unexplored

```

+00  0x55,
+01  0x48, 0x89, 0xe5,
+04  0x48, 0x83, 0xec, 0x10,
+08  0xc7, 0x45, 0xfc, 0x00, 0x00, 0x00, 0x00,
+0f  0x89, 0x7d, 0xf8,
+12  0x48, 0x89, 0x75, 0xf0,
+16  0x8b, 0x7d, 0xf8,
+19  0x48, 0x8b, 0x75, 0xf0,
+1d  0xe8, 0xcc, 0xcc, 0x02, 0x00,
+22  0x48, 0x83, 0xc4, 0x10,
+26  0x5d,
+27  0xc3
+28  0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90

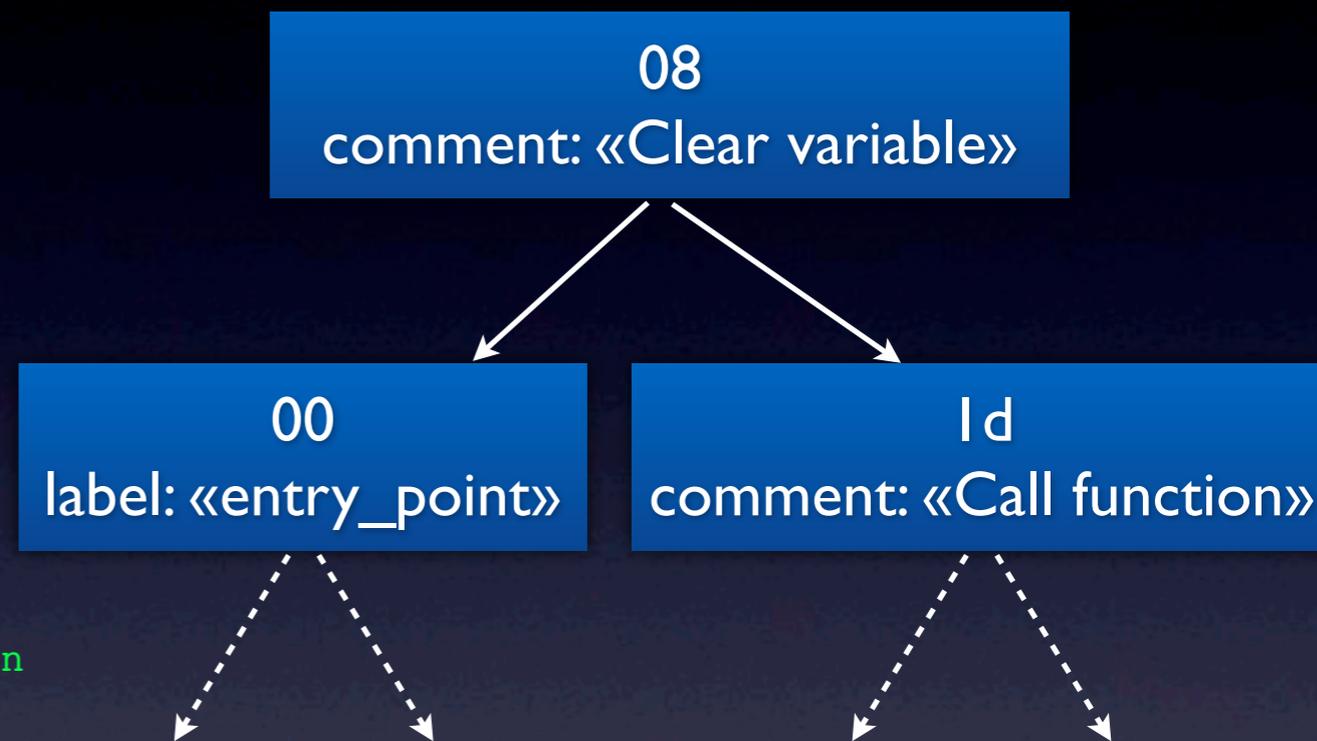
```

Adresse	Type
00	Code
01	Code
02	Next
03	Next
04	Code
05	Next
06	Next
07	Next
08	Code
09	Next
0a	Next
⋮	Next

```

entry_point:
+00  0x55,
+01  0x48, 0x89, 0xe5,
+04  0x48, 0x83, 0xec, 0x10,
      // Clear variable
+08  0xc7, 0x45, 0xfc, 0x00, 0x00, 0x00, 0x00,
+0f  0x89, 0x7d, 0xf8,
+12  0x48, 0x89, 0x75, 0xf0,
+16  0x8b, 0x7d, 0xf8,
+19  0x48, 0x8b, 0x75, 0xf0,
+1d  0xe8, 0xcc, 0xcc, 0x02, 0x00,    // Call function
+22  0x48, 0x83, 0xc4, 0x10,
+26  0x5d,
+27  0xc3
+28  0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90, 0x90

```



```

entry_point:
+00 55          push    rbp
+01 4889E5      mov     rbp, rsp
+04 4883EC10    sub     rsp, 0x10
                // Clear variable
+08 C745FC00000000 mov    dword [ss:rbp+0xffffffffc], 0x0
+0f 897DF8      mov    dword [ss:rbp+0xffffffff8], edi
+12 488975F0    mov    qword [ss:rbp+0xffffffff0], rsi
+16 8B7DF8      mov    edi, dword [ss:rbp+0xffffffff8]
+19 488B75F0    mov    rsi, qword [ss:rbp+0xffffffff0]
+1d E8CCCC0200   call   0x2dcee // Call function
+22 4883C410    add    rsp, 0x10
+26 5D          pop    rbp
+27 C3          ret

```

```
; Basic Block Input Regs: rax rdi - Killed Regs: rbx rsi
```

```
methImpl_AppDelegate_check:
```

```
push    rbp
mov     rbp, rsp
push    rbx
push    rax
mov     rbx, rdi
mov     rsi, qword [0x10004a390]
call   qword [ds:0x100042080]
test   rax, rax
jne    0x10000220d
```

```
-----
; Basic Block Input Regs: rax - Killed Regs: rax
```

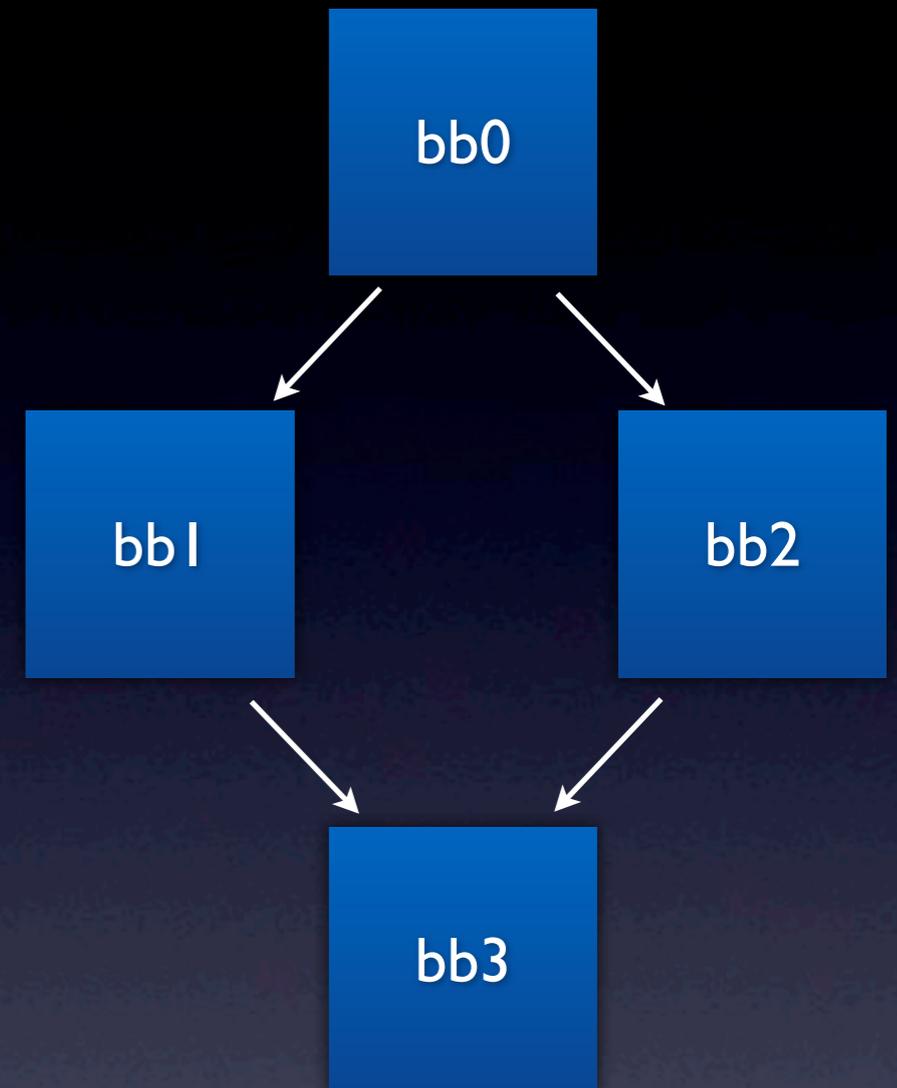
```
xor    al, al
jmp    0x100002220
```

```
-----
; Basic Block Input Regs: rax rbx - Killed Regs: rdx rsi rdi
```

```
mov     rsi, qword [0x10004a3a8]
mov     rdi, rbx
mov     rdx, rax
call   qword [0x100042080]
```

```
-----
; Basic Block Input Regs: rax - Killed Regs: rax rbx rsp rbp
```

```
movsx  eax, al
add    rsp, 0x8
pop    rbx
pop    rbp
ret
```



```

methImpl_AppDelegate_check:
    push    rbp
    mov     rbp, rsp
    push   rbx
    push   rax
    mov    rbx, rdi
    mov    rsi, qword [0x10004a390]           // @selector(registrationData)
    call   qword [ds:0x100042080]          // objc_msgSend
    test   rax, rax
    jne    0x10000220d

    xor    al, al
    jmp    0x100002220

    mov    rsi, qword [0x10004a3a8]           // @selector(checkRegistrationLicense:)
    mov    rdi, rbx
    mov    rdx, rax
    call   qword [0x100042080]          // objc_msgSend

    movsx  eax, al
    add    rsp, 0x8
    pop    rbx
    pop    rbp
    ret

```

```
dec     eax
cmp     eax, 11
ja      0x10000bde0

lea     rcx, qword [ds:0x10000be98] // Jump table
movsxd rax, dword [ds:rcx+rax*4]
add     rax, rcx
jmp     rax // Switch statement with 12 cases
```

Décompilateur

===== B E G I N O F P R O C E D U R E =====

```

; Basic Block Input Regs: rax rdx - Killed Regs: rbx rsi
methImpl_SWFFile_initWithParent_:
00000001000094c4 55          push     rbp
00000001000094c5 4889E5     mov     rbp, rsp
00000001000094c8 53          push     rbx
00000001000094c9 50          push     rax
00000001000094ca 4889D3     mov     rbx, rdx
00000001000094cd 488B35E40F0400  mov     rsi, qword [ds:objc_sel_init] ; @selector(init)
00000001000094d4 FF15A68B0300  call    qword [ds:imp___got__objc_msgSend]
00000001000094da 4885C0     test    rax, rax
00000001000094dd 740B       je      0x1000094ea
; Basic Block Input Regs: rbx - Killed Regs: rax rcx
00000001000094df 488B0D926C0400  mov     rcx, qword [ds:0x100050178]
00000001000094e6 48891C08     mov     qword [ds:rax+rcx], rbx
; Basic Block Input Regs: <nothing> - Killed Regs: rbx rsp rbp
00000001000094ea 4883C408     add     rsp, 0x8 ; XREF=0x1000094dd
00000001000094ee 5B          pop     rbx
00000001000094ef 5D          pop     rbp
00000001000094f0 C3          ret
; endp
```

===== B E G I N O F P R O C E D U R E =====

```
                                ; Basic Block Input Regs: rax rdx - Killed Regs: rbx rsi
                                methImpl_SWFFile_initWithParent_:
00000001000094c4 55                push    rbp
00000001000094c5 4889E5           mov     rbp, rsp
00000001000094c8 53                push    rbx
00000001000094c9 50                push    rax
00000001000094ca 4889D3           mov     rbx, rdx
00000001000094cd 488B35E40F0400  mov     rsi, qword [ds:objc_sel_init] ; @selector(init)
00000001000094d4 FF15A68B0300    call   qword [ds:imp___got__objc_msgSend]
00000001000094da 4885C0           test   rax, rax
00000001000094dd 740B            je     0x1000094ea
                                ; Basic Block Input Regs: rbx - Killed Regs: rax rcx
00000001000094df 488B0D926C0400  mov     rcx, qword [ds:0x100050178]
00000001000094e6 48891C08         mov     qword [ds:rax+rcx], rbx
                                ; Basic Block Input Regs: <nothing> - Killed Regs: rbx rsp rbp
00000001000094ea 4883C408         add     rsp, 0x8 ; XREF=0x1000094dd
00000001000094ee 5B                pop     rbx
00000001000094ef 5D                pop     rbp
00000001000094f0 C3                ret
                                ; endp
```

```
function methImpl_SWFFile_initWithParent_ {
    rbx = rdx;
    rax = [rdi init];
    if (rax != 0x0) {
        *(rax + *0x100050178) = rbx;
    }
    return rax;
}
```

```
mov     r15, rdi
mov     r12, qword [ds:0x1000501a0]
add     r12, r15
mov     r13, qword [ds:0x100050198]
add     r13, r15
xor     ebx, ebx
lea     r14, qword [0x10004b030]

mov     rdi, qword [ds:r13+rbx*8+0x0]
test    rdi, rdi
je      0x100009542
```

```
mov     r15, rdi
mov     r12, qword [ds:0x1000501a0]
add     r12, r15
mov     r13, qword [ds:0x100050198]
add     r13, r15
xor     ebx, ebx
lea     r14, qword [0x10004b030]

mov     rdi, qword [ds:r13+rbx*8+0x0]
test    rdi, rdi
je      0x100009542
```



```
r15 = rdi
r12 = * (int *)0x1000501a0
r12 = r12 + r15
r13 = * (int *)0x100050198
r13 = r13 + r15
ebx = ebx ^ ebx
r14 = 0x10004b030

rdi = * (int *) (r13 + rbx * 8)
CMP(rdi & rdi, 0)
if (EFLAGS & E) goto 0x100009542
```

```
def r15      use rdi
def r12
def r12      use r12, r15
def r13
def r13      use r13, r15
def ebx      use ebx
def r14

def rdi      use r13, rbx
def Z        use rdi
            use Z
```

```
r15 = rdi
r12 = * (int *)0x1000501a0
r12 = r12 + r15
r13 = * (int *)0x100050198
r13 = r13 + r15
ebx = ebx ^ ebx
r14 = 0x10004b030

rdi = * (int *) (r13 + rbx * 8)
CMP(rdi & rdi, 0)
if (EFLAGS & Z) goto 0x100009542
```

<code>def r15</code>	<code>use rdi</code>	<code>r15 = rdi</code>	<code>0</code>
<code>def r12</code>		<code>r12 = * (int *)0x1000501a0</code>	<code>1</code>
<code>def r12</code>	<code>use r12, r15</code>	<code>r12 = r12 + r15</code>	<code>2 r12 instr 1, r15 instr 0</code>
<code>def r13</code>		<code>r13 = * (int *)0x100050198</code>	<code>3</code>
<code>def r13</code>	<code>use r13, r15</code>	<code>r13 = r13 + r15</code>	<code>4 r13 instr 3, r15 instr 0</code>
<code>def ebx</code>	<code>use ebx</code>	<code>ebx = ebx ^ ebx</code>	<code>5</code>
<code>def r14</code>		<code>r14 = 0x10004b030</code>	<code>6</code>
<code>def rdi</code>	<code>use r13, rbx</code>	<code>rdi = * (int *) (r13 + rbx * 8)</code>	<code>7 r13 inst 4, rbx instr 8</code>
<code>def Z</code>	<code>use rdi</code>	<code>CMP(rdi & rdi, 0)</code>	<code>8 rdi instr 7</code>
	<code>use Z</code>	<code>if (EFLAGS & Z) goto 0x100009542</code>	<code>9 Z instr 8</code>

```
def r15      use rdi
def r12
def r12      use r12, r15
def r13
def r13      use r13, r15
def ebx      use ebx
def r14

def rdi      use r13, rbx
def Z        use rdi
```

```
r15 = rdi
r12 = * (int *)0x1000501a0
r12 = r12 + r15
r13 = * (int *)0x100050198
r13 = r13 + r15
ebx = ebx ^ ebx
r14 = 0x10004b030

rdi = * (int *) (r13 + rbx * 8)
if (rdi == 0) goto 0x100009542
```

```
0
1
2  r12 instr 1, r15 instr 0
3
4  r13 instr 3, r15 instr 0
5
6
7  r13 inst 4, rbx instr 8
8  rdi instr 7
```

```
r12 = * (int *)0x1000501a0 + rdi  
r13 = * (int *)0x100050198 + r15
```

```
ebx = 0  
r14 = 0x10004b030
```

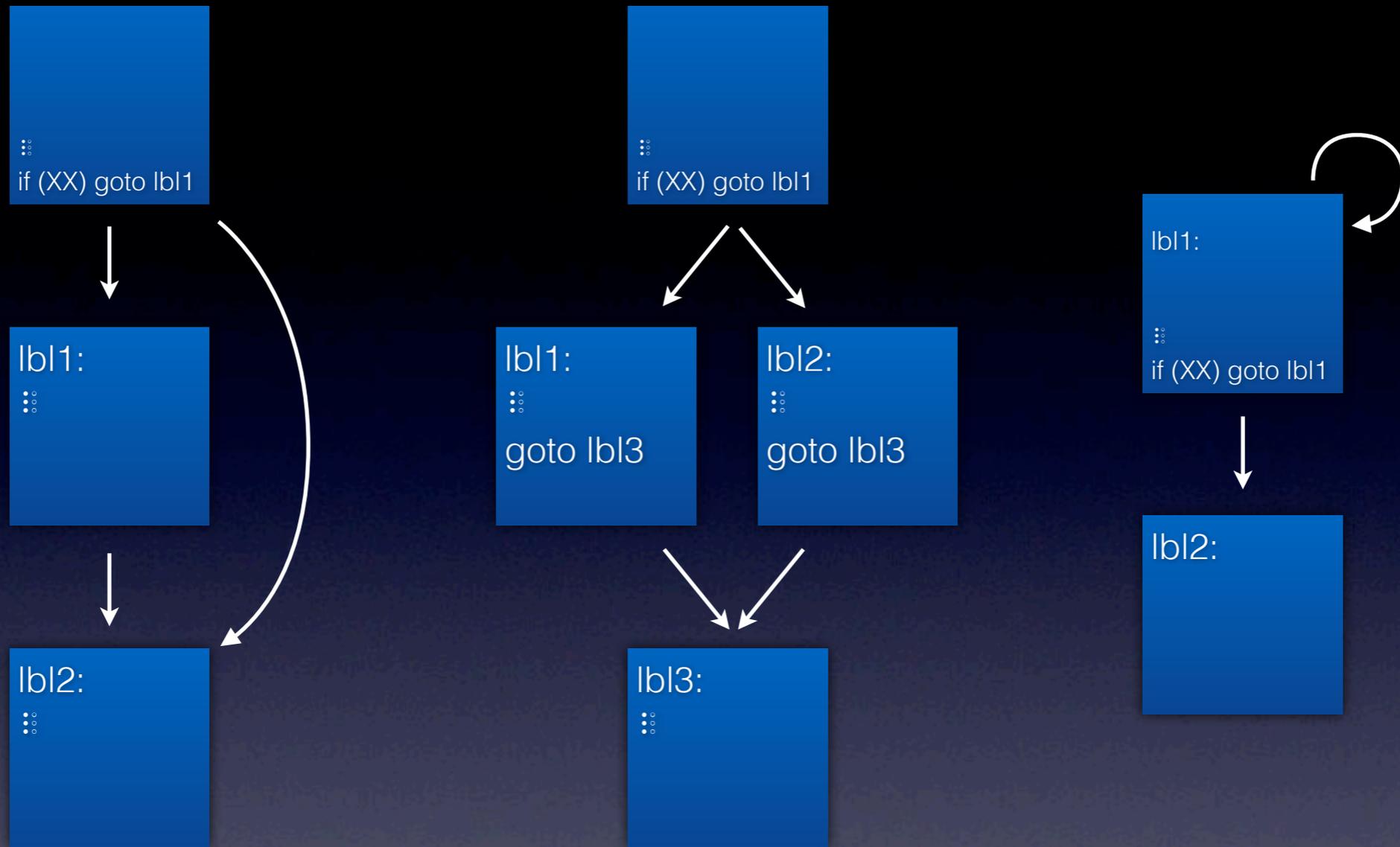
```
if (* (int *) (r13 + rbx * 8) == 0) goto 0x100009542
```

Intel 32bits

```
push eax
push ebx
mov ecx, eax
add ecx, edx
push ecx
call 0xc478f
```

Intel 64bits

```
mov rdi, 0x1
xor rsi, rsi
call 0x10001276
```



```
if (cond) {
    ...
}
```

```
if (cond) {
    ...
} else {
    ...
}
```

```
do{
    ...
}while (cond);
```

```
rax = objc_msgSend(rdi, 0x10000a76e, rdx, rcx);
```

```
rax = objc_msgSend(rdi, 0x10000a76e, rdx, rcx);
```



```
rax = [rdi insertObject:rdx atIndex:rcx];
```

Debugger

- ▶ Serveur GDB
 - ▶ debuggage à distance,
 - ▶ architecture différente (futur)
- ▶ Interface simple pour l'intégration d'autres debuggers (futur)

GDB

▶ || ⏴ ⏵ ↺ ■ — *Signaled (Signal 2 = SIGINT)*

Threads

Thread 1: WAITING

Callstack

```

00007FFF5FC0D6E6 - dyld ZL18gdb ima...
00007FFF5FC0D2CC - dyld Z9notifyGDB...
00007FFF5FC049B4 - dyld ZN4dyldL18n...
00007FFF5FC0EFC4 - dyld ZN11ImageLo...
00007FFF5FC04CE4 - dyld ZN4dyld4lin...
00007FFF5FC05A56 - dyld ZN4dyld5 ma...
00007FFF5FC01397 - dyld ZN13dyldboo...
00007FFF5FC0105E - dyld dyld start

```

GPR		FPU	MMX	Memory	GDB Console		
RAX	0000000000000000	⊖	R8	0000000000000008	⊖	CS	0000002B
RBX	000000000000009D	⊖	R9	00000000000004E8	⊖	DS	00000000
RCX	00007FFF5FBFE140	⊖	R10	00007FFF94B1C000	⊖	ES	00000000
RDX	00007FFF5FBFE140	⊖	R11	00007FFF5FC46C4C	⊖	FS	00000000
RSI	000000000000009D	⊖	R12	000000000000009D	⊖	GS	00000000
RDI	0000000000000000	⊖	R13	0000000000000014	⊖	SS	00000000
RBP	00007FFF5FBFE130	⊖	R14	0000000000000000	⊖	EFLAGS	00000246
RSP	00007FFF5FBFE110	⊖	R15	00007FFF5FC354B8	⊖		
RIP	00007FFF5FC0D6E6	⊖					

O D I T S Z A P C

La suite...

- ▶ plugins
 - ▶ autres types de debugger,
 - ▶ autres processeurs,
 - ▶ fonctionnalités...
- ▶ travail collaboratif
 - ▶ CrowdRE
- ▶ propagation de type
- ▶ amélioration de l'interface
 - ▶ système de tag,
- ▶ ...

<http://www.hopperapp.com/>

 bsr43@hopperapp.com

 [@bSr43](#)