目录

前言	1.1
Xcode调试概览	1.2
Xcode调试心得	1.3
调试区	1.3.1
快捷键	1.3.2
断点	1.3.3
条件判断	1.3.3.1
给汇编加断点	1.3.3.2
调试中	1.3.4
日志输出	1.3.4.1
函数调用堆栈	1.3.4.2
断点触发详情	1.3.4.3
停下来的原因	1.3.4.4
箭头指向的指令	1.3.4.5
切换进程视图	1.3.4.6
Report Navigator	1.3.5
Build Phases	1.3.6
其他	1.3.7
附录	1.4
参考资料	1.4.1

### Xcode开发:调试心得

- 最新版本: v0.6
- 更新时间: 20221031

### 简介

此处介绍Xcode开发中调试方面的心得。先是Xcode调试的概览;再介绍调试心得,包括调试区的介绍、调试快捷键、断 点、调试中、Report Navigator、Build Phases;断点包括条件断点、给汇编加断点;调试中包括日志输出、函数调用堆栈、 断点触发详情、代码停下来的原因、箭头指向的指令、切换进程视图。

### 源码+浏览+下载

本书的各种源码、在线浏览地址、多种格式文件下载如下:

### HonKit源码

• crifan/xcode\_dev\_debug\_summary: Xcode开发:调试心得

### 如何使用此HonKit源码去生成发布为电子书

详见: crifan/honkit\_template: demo how to use crifan honkit template and demo

### 在线浏览

- Xcode开发:调试心得 book.crifan.org
- Xcode开发:调试心得 crifan.github.io

### 离线下载阅读

- Xcode开发:调试心得 PDF
- Xcode开发:调试心得 ePub
- Xcode开发:调试心得 Mobi

### 版权和用途说明

此电子书教程的全部内容,如无特别说明,均为本人原创。其中部分内容参考自网络,均已备注了出处。如发现有侵权,请通过邮箱联系我 admin <sup>艾特</sup> crifan.com,我会尽快删除。谢谢合作。

各种技术类教程,仅作为学习和研究使用。请勿用于任何非法用途。如有非法用途,均与本人无关。

### 鸣谢

感谢我的老婆陈雪的包容理解和悉心照料,才使得我 crifan 有更多精力去专注技术专研和整理归纳出这些电子书和技术教 程,特此鸣谢。

### 更多其他电子书

本人 crifan 还写了其他 150+ 本电子书教程, 感兴趣可移步至:

### crifan/crifan\_ebook\_readme: Crifan的电子书的使用说明

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 15:08:45

# Xcode调试概览

在涉及到Apple苹果相关的开发,不论是

- 不同端
  - Mac
  - iOS
- 不同开发语言
  - ObjC
  - Swift
- 不同方向
  - 正向:安全防护
  - 逆向: 破解

往往都要用到Apple的IDE: xcode

而此处整理Xcode开发期间中,涉及到调试方面的心得。

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 15:02:50

# Xcode调试心得

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 10:30:31

## 调试区

对于Xcode调试期间的界面,一般长这样:

•••	• •	youtube master		routube 🔪 📋 iPhone7_1331				▲ 261	+ 🖽
E 🛛 🖬 ۹ Δ 🗇 🍼		BB I < > I m youtubeCronet.xm I	m youtubeReqResp.xm	SRLRegistry_sharedInstance.coffee	0 mach_msg_trap	m youtubeCommon.xm	h youtubeCommon.h		
v 💶 voutube PID 280	00	😑 youtube 👌 🕕 Thread 1 👌 📶 0 mach_ms	g_trap						< 🛆 >
CPU		1 libsystem_kernel.dylib`mach 2 0x18841c62c <+0>: mov	_msg_trap: x16, #-0x1f						
Remory		3 0x18841c630 <+4>: svc 4 -> 0x18841c634 <+8>: ret						= Thread 1: sig	anal SIGSTOP
Finergy Impact									
E Disk									
Network     FPS									
(1) Thread 1 Queue: com.appleain-	thread (serial)								
0 mach_msg_trap									
💼 8 start									
> (i) Thread 2	21								
> (i) Thread 4	<i>.</i> ,								
> (i) Thread 5									
> () Thread 6	- (7)								
<ul> <li>DavaScriptCore bralloc scavengel</li> <li>WebThread (8)</li> </ul>	r (/)								
> () com.apple.uikit.eventfetch-thread	l (9)								
> () Thread 10									
> (i) Thread 11									
> II) Thread 13									
> (I) Thread 14									
> 0 AVAudioSession Notify Thread (15									
> 0 com.google.ios.ssdp (16)	daa (17)								
> (I) Thread 18		■ ID ☆ ¥ 1 0 ≫ 8	n 🔤 youtube 🕽 🕕	Thread 1 > 🔝 0 mach_msg_trap	THE DESKDOTHE	stops only for the thread			ine: 5 Col: 1   🔒
> () Thread 19					argument.				
<ul> <li>Thread 20</li> <li>com.apple.CFSocket.private (21)</li> </ul>					-u <column> (colum Specifies the co</column>	mn <column> ) olumn number on which to</column>	set this breakpoint.		
			De	ebug Area=调词	-v <none> (structu The value for th implementation of more than once.</none>	ured-data-value <none> ) he previous key in the pa of a scripted breakpoint.</none>	ir passed to the Pairs can be specifi	ed	
					-w <boolean> (on-1 Set the breakpoi</boolean>	throw <boolean> ) int on exception throW.</boolean>			
					-x <thread-index> ( - The breakpoint : argument.</thread-index>	thread-index <thread-in stops only for the thread</thread-in 	dex> ) whose index matches 1	his	
					-y <linespec> (jo A specifier in t line breakpoints</linespec>	int-specifier <linespec> the form filename:line[:c s.</linespec>	) olumn] for setting fi]	* *	
💿 Filter	🖬 🖬 🖬	Auto 0 🕕 🕕		Filter	II Output 0			© Filter	

其中右下角的区域是Xcode的 debug区 = 调试区

此处对于 xcode调试区 的 按钮 = 功能 进行概述:



crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 15:06:28

## 快捷键

### Xcode调试相关快捷键

- Xcode调试时的相关快捷键
  - 。 文字
    - Step Over : F6
      - Step Over Instruction : Ctrl+F6
      - Step Over Thread : Ctrl+Shift+F6
    - Step Into : F7
      - Step Into Instruction : Ctrl+F7
      - Step Into Thread : Ctrl+Shift+F7
    - Step Out : F8

o 图

Debug	Source Control	Window	Help						
Continue			^%Y						
Continue	To Current Line		^#C						
Step Ove	Step Over								
Step Into			F7						
Step Out			F8						
Step Ove	er Instruction		^F6						
Step Ove	er Thread		^샵F6						
Step Into	Instruction		^F7						
Step Into	Thread		^습F7						
Deactiva	te Breakpoints		жY						
Breakpoi	nts		>						
Debug W	/orkflow		>						
Attach to	Process by PID o	r Name							
Attach to	Process		>						
Detach fi	rom youtube								
Debug Ex	xecutable								
Capture	GPU Workload								
Simulate	Location		>						
Simulate	Background Fetc	h							
Simulate	MetricKit Payload	ls							
Simulate	UI Snapshot								
View Deb	ougging		>						
StoreKit			>						

## F7无效

Mac中此处Xcode调试的单步进入的快捷键: F7 无效

发现是被其他占用了。

经过查找发现是旧版有道词典占用的。

解决办法:

- 彻底卸载旧版有道词典
  - 。因为即使没开启取词和划词也会占用F7快捷键
- 卸载旧版,安装新版有道词典
  - 。 没开启取词划词,就不会占用(F7等)快捷键

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 10:46:46

# 断点

### 调试加断点时函数名要准确,否则加不上

#### 想要给函数

\_\_\_\_lldb\_unnamed\_symbol162

#### 加上断点:

Ś	Xcode	File	Edit	View	Find	N	laviga	ate	Edit	or	Prod	uct	Debug	Source Control	Window	w Help
••	•														•	🛃 Aweme
				8	$\times$	П	Q	≙	$\bigcirc$	ð	•	=				器   < >
✓ ▲ Aweme 2 Breakpoints (1 disabled)											No Selection					
<ul> <li>&gt; Z _dyld_register_func_for_add_image</li> <li>2 dyld3::_dyld_register_func_for_add_image(void (*)(mach_header const*, long)) libdyld.dylib</li> <li>2 _dyld_register_func_for_add_image libdyld.dylib</li> </ul>																
	Σ <u>    lldb</u>	_unnan	ned_syn	nbol162												

后来发现,前面的写法是错误的

应该改为:

\_\_\_lldb\_unnamed\_symbol162\$\$AwemeCore

不过然后又发现,好像没有生效

且对于为何没有生效,还有额外的提示:

Xcode won't pause at this breakpoint because it has not been resolved. Resolving it requires that: The symbolic name is spelled correctly. The symbol actually exists in its library. The library for the breakpoint is loaded.

🗰 Xcode File Edit View Find Navigate Editor I	Product Debug Source Control Window Help
	Aweme Aweme IPhone7_1331
	Barrier Contraction Contractio
- 🛃 Aweme 2 Breakpoints (1 disabled)	Aweme ) Thread 1 in 0_dyld_register_func_for_add_image
<ul> <li>Z _dyld_register_func_for_add_image ID 1</li> <li>Z dyld3::_dyld_register_func_for_add_image(void (*)(m</li> <li>Z _dyld_register_func_for_add_image ID 1.2 libdyld.dylib</li> </ul>	2 -> 0x1bdb57ef4 <+0>: sub sp, sp, #0x30 3 0x1bdb57ef8 <+4>: stp x20, x19, [sp, # 4 0x1bdb57efc <+8>: stp x29, x30, [sp, #
Idb_unnamed_symbol162\$\$AwemeCore ID 3	5 0x1bdb57f00 <+12>: add x29, sp, #0x20 6 0x1bdb57f04 <+16>: mov x19, x0
	Xcode won't pause at this breakpoint because it       Irp       x8, 287966         has not been resolved.       Id       x8, x8, #0x6a8         Resolving it requires that:       Irb       w8, [x8]         • The symbol crame is spelled correctly.       iz       w8, 0x1bdb57f2c         • The library for the breakpoint is loaded.       Ip       x29, x30, [sp, #         Ip       x20, x19, [sp, #
	14 0x1bdb57f24 <+48>: add sp, sp, #0x30 15 0x1bdb57f28 <+52>: b 0x1bdb63fdc (*)(mach_header const*, long))

再然后发现是:

是函数所属的二进制弄错了

#### 应该改为:

\_\_\_lldb\_unnamed\_symbol162\$\$Aweme

•••	I										
			$\times$	Π	Q	≙	$\Diamond$	ð		<b>=</b>	
<ul> <li>Awe</li> <li>2 _</li> <li>2</li> <li>4</li> </ul>	eme 3 Breakpoints dyld_register_func_for_a dyld3::_dyld_register_fu _dyld_register_func_fou _lldb_unnamed_symbol logos_function\$_ungrou	add_imag unc_for_ r_add_in 162\$\$Av ped\$_dy	ge ID add_ nage weme /ld_re	4 ID 4 ID 4 ID 5	ge(vo I.2 lik 6.1 er_fu	oid (* odyld unc_f	) (mad .dylik or_ad	ch_h ) dd_in	eader	r const*, long)) ID 4.1 libdyld.dylib ID 5.1	

就正确了。

后续断点才能正常生效:

	🛃 Aweme	Aweme > iPhone7_1331	Running Aweme on iPhone7_1331 🔒 5 🕂 📑
🖻 🛛 🗔 Q. 🛆 🗇 🧃 🗆 🗐	⊞   < >   E hook_misc.xm	0lldb_unnamed_symbol162\$\$Aweme	E0 - 🔁
Aweme PID 33829     O	Aveme ) Thread 1 ) 1 0 1 1 Aveme ) Thread 1 ) 1 0 1 1 Aveme ) 1 1 1 0 1 1 0 1 3 0 x1860756c <402 4 0 x1860756d <402 5 0 x1860756d <422 6 0 x1860756d <424 9 0 x1860756e <428 1 0 x1860756e <428 1 0 x1860756e <428 1 0 x18607576e <428 1 0 x1860756e	Jdb_unnamed_symbol162\$\$Aweme           d_symbol162\$\$Aweme:           : stp x22, x21, [sp, #=0x30]!           : stp x24, x19, [sp, #=0x30]!           : stp x29, x38, [sp, #0x20]           : add x29, sp, #0x20]           : add x29, sp, #0x20]           : add x29, sp, #0x20]           : add x1, x1, #0x640           : mov w0, #0x1           : bl 0x1ad2y040*0           : bl 0x1ad2y040*0           : bl 0x1ad2y040*0           : add x20, x1260*0766c           : add x2, y6, #0x20	Thread 1: breakpoint 2.1 (1) =0x20 =0x640lldb_unnamed_symbol1202227\$\$Aweme <+160>
<ul> <li>Thread 1 Queue: com.apple.main-thread (serial)</li> <li>0lldb_unnamed_symbol162\$\$Aweme</li> <li>1 _logos_meta_method\$_ungrouped\$_RxAnnotationInli</li> <li>2 load_images</li> <li>10_dyid_start</li> <li>Thread 2</li> <li>Thread 3</li> </ul>	2 0x185607514 < <442 3 0x185607515 < <443 10 0x185607515 < <443 10 0x185607608 <<52 10 0x185607608 <<56 10 0x185607608 <<66 10 0x185607618 <<66 10 0x185607618 <<76 20 0x18507618 <<76 20 0x18507618 <<76 20 0x18507618 <<76 20 0x18507618 <<76 20 0x18507618 <<76 20 0x18507618 76<br 20 0x18507618 <<76 20 0x18507618 76<br 20 0x185076180	<ul> <li>idt X8, (X8, mexes)</li> <li>idtr X1, (X8, Mexes)</li> <li>idtr X1, (X8, Mexes)</li> <li>idtr X1, (X8, Mexes)</li> <li>idtr X9, -398</li> <li>idtr X8, [X9, Mex648]</li> <li>istr X9, X8</li> </ul>	:lldb_unnamed_symboll202162\$\$Aweme :lldb_unnamed_symboll202160\$\$Aweme
<ul> <li>Intread 4</li> <li>Thread 5</li> <li>Thread 6</li> <li>Thread 7</li> </ul>		Average Series of the serie	0
🕞 Filter 🔳 🖬 🖬	Auto 🌣 🛛 💿 🧻 🖲 Filter	All Output ≎	🕲 Filter

### 添加符号断点时,会自动搜索到匹配的函数

#### 给Xcode添加符号断点:

\_dyld\_get\_image\_name

<ul> <li>sysctl ID 25.1</li> <li>S_dyld_get_image_name ID 26</li> </ul>		187 188 %hookf(int	:, UIApplicat ⊥ © ≫
<ul> <li>dyld3::_dyld_get_image_name(unsigned int) IE</li> <li>_dyld_get_image_name ID 26.2 libdyld.dylib</li> </ul>	Enable Symbolic Breakpoint          Name         A breakpoint name cannot start with nu         Symbol      dyld_get_image_name         Module       Executable or library name         Condition       Ignore         Ignore       © times before stopping         Action       Add Action         Options       Automatically continue after end	mbers or contain any white	* space. * *) 7ef
+ 🕞 Filter		Auto 🌣 🛛 💿 👔	Filter
ー 回车确认后发现: 自动会出现2个(子)断点:			

- dyld3::\_dyld\_get\_image\_name(unsigned int) ID 26.1 libdyld.dylib
- \_dyld\_get\_image\_name ID 26.2 libdyld.dylib

Sysctl ID 25.1

- Σ\_dyld\_get\_image\_name ID 26
  - D dyld3::\_dyld\_get\_image\_name(unsigned int) ID 26.1 libdyld.dylib
  - S\_dyld\_get\_image\_name ID 26.2 libdyld.dylib

看起来好像是:

会根据当前符号,自动去寻找匹配到的函数

效果不错。

自动补全

Xcode中添加断点的输入框中,竟然也支持自动补全

折腾:

【未解决】研究抖音越狱检测逻辑:NSObject的load

期间,发现个心得:

Xcode的符号断点的输入框中,也支持 动态匹配 自动补全:

<ul> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>method_excha</li> <li>method_setIm;</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>+[NSObject load]</li> </ul>	<pre>C Enable Symbolic Breakpoint Name A breakpoint name cannot start with numbers or contain any white space. Symbol +[NS] Modu NSObject NSURL Conditic NSDictionary Igno NSData Actic NSArray NSNumber NSString NSInteger</pre>
+ 🖲 Filter	Auto 🌣 🛛 💿 👔 🐨 Filter
<ul> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>method_excha</li> <li>method_setImp</li> <li>Ildb_unnam</li> <li>Ildb_unnam</li> <li>+[NS</li> </ul>	Enable Symbolic Breakpoint          Name       A breakpoint name cannot start with numbers or contain any white space.         Symbol       +[NSObject           Module       log         Module       log         Condition       Logos         Ignore       0         Action       Add         Id       lu         Options       Auto         Iib       lib
+ (⑦ Filter 输入完后的效果:	Auto 2 ( ) (i) Filter

lldb_unnam		A breakpoint name cannot start with numbers or contain any white space.
lldb_unnam	Symbol	+[NSObject load]
lldb_unnam	Module	Executable or library name
IIdb_unnam	Condition	
method_excha	Ignore	0 🗘 times before stopping
method_setIm	Action	Add Action
lldb_unnam	Options	Automatically continue after evaluating actions
lldb_unnam⊾.		
+[NSObject load		

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 14:57:42

# 条件判断

TODO:

- 【已解决】XCode中如何给符号断点加上判断条件
- 【未解决】通过XCode给stringByAppendingString加断点调试寻找抖音崩溃原因
- 【已解决】XCode调试抖音ipa: 给用Logos去hook的函数\_dyld\_get\_image\_name加符号断点
- 举例
  - Symbol: \_dy1d\_get\_image\_name
    - Condition: (\$arg1 == 0) || (\$arg1 == 1)



crifan.org,使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 14:11:37

### 给汇编加断点

TODO:

• 【未解决】Xcode中给iOS的ObjC的ARM汇编代码加上带条件判断执行的断点

#### 给汇编加断点

调试二进制 = ARM汇编代码,可以给汇编加上断点:

其实本身很简答:点击每行的最左边,即可加断点

只不过有个小缺点,有时候是个大缺点:由于断点开启后,该行最左边会加上白色背景,但是本身界面就是白色背景,导致:你根本不知道自己加了断点

只有仔细看,才能看出来:此时的行号没了

然后此时右键,能看到断点相关菜单功能,也才能,确认的确加了断点

举例:

点击行号 13, 可以加上 汇编代码的断点:

此时 13 的行号, 就看不见了:

•	🛃 Aweme		💮 Aweme 👌 📋 iPhone7P_1341 🛛 🛛 🕅						Running Aweme on iPhone7P_1341 🔒 14 +			+	
	BB   < >	AwemeDylil	b.xm	E Aweme	ylib.xm	0_dyld_ge	tim	age_name					=0
D	Aweme > ()	Thread 1 $ angle$ 🛅	0 _dyld_get	_image_nam	e								< .
	1 libdy	ld.dylib`_dy	ld_get_i	mage_name	:								
0	2 -> 01	x1ad825518 <	+0>: s	tp x20	, x19, [sp	#-0x20]!							
D	3 0)	x1ad82551c <	+4>: s	tp x29	, x30, [sp	#0x10]							
В	4 0:	x1ad825520 <	+8>: a	dd x29	, sp, #0x10	3	;	=0x10					
0	5 0:	x1ad825524 <	+12>: m	ov x19	, x0								
0	6 0:	x1ad825528 <	+16>: a	drp x8,	303304								
s	7 0:	x1ad82552c <	+20>: a	dd x8,	x8, #0x698	3	;	=0x698					
	8 0:	x1ad825530 <	+24>: 10	drb w8,	[x8]								
S	9 0:	x1ad825534 <	+28>: cl	bz w8,	0x1ad82554	48	;	<+48>					
	10 0:	x1ad825538 <	+32>: m	ov x0,	x19								
	11 0:	x1ad82553c <	+36>: 10	dp x29	, x30, [sp	#0x10]							
	12 0:	x1ad825540 <	+40>: 10	dp x20	, x19, [sp]	, #0x20							
		x1ad825544 <	+44>: b	0x1	ad830924		;	dyld3::_	dyld_get	_image_n	ame(uns:	igned int)	
1	14 0	x1ad825548 <	+48>: a	drp   x8,	315596						= Thr	ead 1: instruct	tion step over
-	15 0:	x1ad82554c <	+52>: 10	dr x1,	[x8, #0x74	48]							
	16 0:	x1ad825550 <	+56>: cl	bnz x1,	0x1ad8255	70	;	<+88>					
			10		00								

但是可以右键发现是加了断点的:

🛃 Awe	Aweme 🔅 Awen			7P_1341		Running Aweme on iPhone7P_1341			
88 <	>   🔄 AwemeDy	lib.xm	🖻 AwemeD	ylib.xm 🕅 0_dy	ld_get_image_nar	ne			
Awerr	ne 👌 🅕 Thread 1 〉 🛅	0 _dyld_get	t_image_nam	e					
1 ]	libdyld.dylib`_d	yld_get_i	.mage_name	:					
2 -	-> 0x1ad825518	<+0>: s	tp x20	x19, [sp, #-0x20	9]!				
3	0x1ad82551c	<+4>: s	tp x29	x30, [sp, #0x10]	]				
4	0x1ad825520	<+8>: a	dd x29	sp, #0x10	; =0x10	)			
5	Øx1ad825524	<+12>: m	10v x19	, x0					
6	Øx1ad825528	<+16>: a	drp x8,	303304					
7	Øx1ad82552c	<+20>: a	dd x8,	x8, #0x698	; =0x69	28			
8	0x1ad825530	<+24>: 1	drb w8,	[x8]					
9	0x1ad825534	<+28>: c	bz w8,	0x1ad825548	; <+48>	•			
10	Øx1ad825538	<+32>: m	10v x0,	x19					
11	Øx1ad82553c	<+36>: 1	.dp x29	x30, [sp, #0x10]	]				
12	0x1ad825540	<+40>: 1	.dp x20	x19, [sp], #0x20	9				
	lik Deselvestet		0x1a	ad830924	; dyld3	<pre>3::_dyld_get_image_name(uns:</pre>	igned int)		
Ed	iit Breakpoint		lrp x8,	315596		= Thr	ead 1: instruction		
En	hable Breakpoint		lr x1,	[x8, #0x748]					
De	elete Breakpoint		inz x1,	0x1ad825570	; <+88>				
			lrp x0,	39					
Re	eveal in Breakpoint	Navigator	ld x0,	x0, #0xe0b	; =0xe0	b			
1-	UNTRADE0000		drp x20	315596					

另外,切换到断点类别中的断点列表,也能看到新的 汇编代码 的断点:

		Awen	me	weme > 📋	iPhone7P_1341	Running Aweme on iPho	ne7P_1341 🔒 14
	88	< >	AwemeDylib.xm	F	AwemeDylib.xm	et_image_name	
✓ ▲ Aweme 16 Breakpoints (2 disabled)	0	Aweme	) 🕕 Thread 1 ) 🧰 0 _d	ld_get_ima	ge_name		
✓ I AwemeDylib.xm		1 li	bdyld.dylib`_dyld_	get_image	e_name:		
Unknown ID 5 1 169		2 ->	0x1ad825518 <+0>	stp	x20, x19, [sp, #-0x20]!		
		3	0x1ad82551c <+4>	stp	x29, x30, [sp, #0x10]	0×10	
[Noothing strips Dutes and in othing in the othing in the other strips of the other		5	0x1ad825526 <+0>	: mov	x19, x0	, =0X10	
-[NSString stringByAppendingString:] ID 2.1		6	0x1ad825528 <+16	: adro	x8, 303304		
awemeMain ID 3		7	0x1ad82552c <+20	: add	x8, x8, #0x698	; =0x698	
2 _awemeMain ID 4		8	0x1ad825530 <+24	: ldrb	w8, [x8]		
VIApplicationMain		9	0x1ad825534 <+28	: cbz	w8, 0x1ad825548	; <+48>	
UIApplicationMain 190		10	0x1ad825538 <+32	: mov	x0, x19		
IllKit IllApplicationMain(Swift Int32 Swift Optional Swift		11	0x1ad82553c <+36	: ldp	x29, x30, [sp, #0x10]		
	=	12	0x1ad825540 <+40	: Tab	x20, x19, [sp], #0x20	. duldle, duld get image	neme(unsigned int)
		1/	0x1ad825544 <+44	adrn	v8 315596	; dyids::_dyid_get_image_	= Thread 1: instructi
+ [AWECloudJailBreakUtility jailbroken] ID 6		15	0x1ad82554c <+52	: ldr	x1, [x8, #0x748]		- medu i. matructi
E + [AWECloudJailBreakUtility init] ID 7		16	0x1ad825550 <+56	: cbnz	x1, 0x1ad825570	; <+88>	
+ [AWECloudJailBreakUtility initialize] ID 8		17	0x1ad825554 <+60	: adrp	x0, 39		
FAWECloudJailBreakUtility alloc ID 9		18	0x1ad825558 <+64	: add	x0, x0, #0xe0b	; =0xe0b	
AWECloud IsilBreak Itility init] ID 10		19	0x1ad82555c <+68	: adrp	x20, 315596		
		20	0x1ad825560 <+72	: add	x20, x20, #0x748	; =0x748	
[2] +[AWECIOUdJaliBreakOtility Initialize] ID 11		21	0x1ad825564 <+76	: mov	x1, x20		
+[AWECloudJailBreakUtility alloc] ID 12		22	0x1ad825568 <+80	DI lata	0x1ad824d38	; _dy1d_Tunc_lookup	
Sysctl ID 13.1		23	0x1a082556C <+84	a mov	XI, [X20]		
✓ ∑ _dyld_get_image_name ID 14		24	0x1ad825576 <+00	: 1dn	x29, x30, [sp. #0x10]		
D dyld3::_dyld_get_image_name(unsigned int) ID 14.1 libd		26	0x1ad825578 <+96	: ldp	x20, x19, [sp], #0x20		
Avid get image name ID 14.2 libdvid dvlib		27	0x1ad82557c <+10	>: br	x1		
		28					
(A) 0x18d020044 1D 17.1	-		. + 1 0 °		🛛 🛛 🌐 Aweme 🕽 🍈 Thread 1 🤇 🛅 C	0_dyld_get_image_name	Line

#### 此处断点名就是:地址:

#### 0x1ad825544

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 14:52:45

# 调试中

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 10:30:31

### 日志输出

### 调试区中log日志输出为空

现象

偶尔遇到过,调试区中,输出log的地方是空白=所有日志都没有输出=看不到日志

● 原因

后来发现是,之前某次不小心,把默认的 All Output 改为了其他的选项了,比如 Debugger Output 或 Target Output



#### • 解决办法

改回默认的 All Output,即可正常看到全部log了。

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 11:00:04

### 函数调用堆栈

#### 点击顶部函数名查看函数调用顺序=函数调用堆栈

#### 无意间发现:

#### 点击调试界面的顶部的 App名字同行的位置的 最后的 函数名:



#### 可以列出: 函数调用顺序 = 函数调用堆栈:

	Aweme	iii Aweme > 🚺	iPhone7P_1341	Running Aweme on iPhone7P_1341	<mark>A</mark> 14	+ 🖪 🗖
🖿 🛛 🗔 🔍 🛆 🗇 🍯 🗖 🗐	88   < >	0_dyld_get_image_name				EO   🕀 👗
Aweme PID 12725      Awem	Image: Non-Section 2         Image: N	O. Jyld. get. Image.name     O. Jyld.get.image.name     O. Jyld.get.image.name     O. Jyld.get.image.name     O. Jyld.ext.image.name     O. Jyld.get.image.name     O. Jyld.get.im	Library mageInit (mageLoader::LinkContext mitialization (mageLoader::LinkConte vitialization (mageLoader::LinkConte vitialization (mageLoader::LinkContext cars(mageLoader::LinkContext const table() der const, unsigned long, int, char	constå) xt constå) tt constå, unsiconst*, ImageLoader::initializerTimin tt constå, unsiconst*, ImageLoader::initializerTimingLi &, ImageLoader::initializerTimingListå) const*, char const**, char const**, unsigned long*)	gList&, ImageLoad gList&, ImageLoad st&, ImageLoader::	er::UninitedUpwards&) ler::UninitedUpwards&) :UninitedUpwards&)
	14 15 16 17	0 12_dyld_start 0 12_dyld_start 0x1ad825554 <+60>: adro	x0. 39	onstra, unsigned iongry		

#### -》方便调试。

#### 类似的,点击 Thread 也可以列出函数:

•••		🛃 Aweme	Aweme > i iPhone7P_1341	Running Aweme on iPhone7P_1341	A 14 + 🖪 🔽
🖿 🛛 🗔 q. 🛆 🗇 🍯 🗖 🗏		器   < >	get_image_name		
Aweme PID 12725      CPU      Memory      III     Memory      III     Kerrey Impact      Disk      Network      Thread 1 Queue: com.apple.main-thread (serial)      O_dyld_get_Image_name	(1)     (2)     (	Aver         Thread 1           1         1bdy1d.dy1b.'           2         >> 0tad25511           3         0tad25521           4         0tad25521           5         0tad25522           6         0tad25522           7         0tad25522           8         0tad25522           9         0tad25532           9         0tad25533           10         0tad25533           11         0tad25533           12         0tad25533           12         0tad25543           13         0tad25544	O_dyld_get_image_name     O_dyld_get_image_name     O_dyld_get_image_name     O_d_CFGetHandleForLoadedLibrary     O_d_CFGetHandleForLoadedLibrary     O_d_CFGetHandleForLoadedLibrary     O_dotAllaria     SimageLoaderMachO::dointialization(imageLoader::     O imageLoader::recursiveInitialization(imageLoader::     O imageLoader::recursiveInitialization(imageLoader::     O imageLoader::recursiveInitialization(imageLoader::     O imageLoader::recursiveInitialization(imageLoader::     O imageLoader::recursiveInitialization(imageLoader::     O imageLoader::recursiveInitializers(imageLoader::LinkCoi     O dyld::initializeMainExecutable()     I 0 dyld::.main(machch_beader const; unsigned long     I 0 dyld::.main(machch_beader const; unsigned long	nkContext const&) ::LinkContext consImageLoader::InitializerTiming LinkContext consImageLoader::InitializerTiming Kontext const&ImageLoader::InitializerTiming text const&, ImageLoader:InitializerTimingList&) , int, char const*, char const*, char const*, uns	Listā, imageLoader::UninitedUpwardsā) Listā, imageLoader::UninitedUpwardsā) Listā, imageLoader::UninitedUpwardsā) igned long*)
12_dyld_start		14         0x1ad825548           15         0x1ad825546           16         0x1ad825556           17         0x1ad825554	11 dyldbootstrap::start(dyld3::MachOLoaded const, 2 _dyld_start <+60>: adrp x0, 39	int, char const**, dyld3::MachOLoaded const*, ur	asigned long*)

#### -》看来是:

方便直接切换到不同的代码执行的地方:

- Арр
  - Thread
    - Function

另外:

调试界面的底部:

直接点击,会显示出:

和调试左上角列出的一样的,只有2个函数:

•••		🔼 Aweme	98	🔋 Aweme 👌 💄	iPhone7P_1341	Running Aweme on iP
🖻 🛛 🗔 Q 🛆 🔶 🍺 🕞 🗐		88   < >	0_dyld_get_i	mage_name		
<ul> <li>✓ I Aweme PID 12725</li> <li>■ CPU</li> <li>■ Memory</li> <li>■ Memory</li> <li>■ Foregy Impact</li> </ul>	<ul> <li>(1) (1) (2) (2) (2) (2) (2) (2) (2) (2) (2) (2</li></ul>	Aweme ) 1 libdy 2 -> 0 3 0 4 0 5 0 6 0	<pre>Thread 1 &gt; iii 0 yld.dylib`_dyl 0x1ad825518 &lt;+ 0x1ad82551c &lt;+ 0x1ad825520 &lt;+ 0x1ad825524 &lt;+ 0x1ad825528 &lt;+</pre>	_dyld_get_image d_get_image 0>: stp 4>: stp 8>: add 12>: mov 16>: adrp	<pre>g_name name: x20, x19, [sp, #-0x20]! x29, x30, [sp, #0x10] x29, sp, #0x10 x19, x0 x8, 303304</pre>	; =0x10
<ul> <li>Disk</li> <li>Network</li> <li>Thread 1 Queue: com.apple.main-thread (serial)</li> <li>0_dyld_get_image_name</li> <li>12_dyld_start</li> </ul>	Zero KB/s Zero KB/s	7     6       8     6       9     10       11     12       13     14       15     16       17     18       19     10       21     22       23     24       25     6	N1ad82552c         0x1ad82552c         0x1ad82552d         0x1ad825538         0x1ad82553c         0x1ad82553c         0x1ad82554c         0x1ad82554d         0x1ad82554c         0x1ad82554c         0x1ad82554c         0x1ad82554c         0x1ad82554c         0x1ad82554c         0x1ad82555c         0x1ad82556c         0x1ad82557c         0x1ad82557c         0x1ad82557c	20::       add         24::       ldrb         24::       ldrb         28::       cbz         36::       ldp         40::       ldp         440::       ldp         440::       ldp         52::       ldr         56::       cbz:         60::       addrp         72::       add         68::       adrp         72::       add         80::       b1         84::       ldr         84::       ldr         84::       ldr         82::       mov         92::       ldp	<pre>x8, x8, #0x698 w8, [x8] w8, 0x1ad825548 x0, x19 x29, x30, [sp, #0x10] x20, x19, [sp], #0x20 0x1ad830924 x8, 315596 x1, [x8, #0x748] x1, 0x1ad825570 x0, x0, #0xe0b x20, 315596 x20, x20, #0x748 x1, x20 0x1ad824d38 x1, [x20] x29, x30, [sp, #0x10]</pre>	<pre>; =0x698 ; &lt;+48&gt; ; dyld3::_dyld_get_imag ; &lt;+88&gt; ; =0xe0b ; =0x748 ; _dyld_func_lookup</pre>
		26 27 28	vx1a0825578 <+ 0x1ad82557c <+ >↓ ↓ ↑ ↑	90>: 1dp 100>: br ≫ 🕃 🖓	7 Aweme ) Thread ( 0 0 (11db ( 12 (11db ( 12))	_dyld_get_image_name _dyld_start

#### 后来发现:

鼠标移动上去,会提示:

Choose stack frame (hold Command to show full backtrace)

13 14 16 16 17 18 <b>19</b> 20 21 21 22 23 24 25 26 26	0x1ad825544           0x1ad825548           0x1ad825544           0x1ad825544           0x1ad825554           0x1ad825556           0x1ad825558           0x1ad825558           0x1ad825568           0x1ad825568           0x1ad825568           0x1ad825568           0x1ad825568           0x1ad825568           0x1ad825568           0x1ad825570           0x1ad825574           0x1ad825574           0x1ad825578           0x1ad825578	<pre>+44&gt;: b +48&gt;: adrp +56&gt;: cbnz +56&gt;: cbnz +66&gt;: adrp +64&gt;: add +68&gt;: adrp 72&gt;: add 76&gt;: mov 80&gt;: b1 84&gt;: ldr 88&gt;: mov 92&gt;: ldp 96&gt;: ldp</pre>	0x1ad830924 x8, 315596 x1, [x8, #4 x1, 0x1ad82 x0, 39 x0, x0, #0 x20, 315596 x20, x20, # x1, x20 0x1ad824d38 x1, [x20] x0, x19 x29, x30, [ x29, x30, [	4 ; dyld3 2x570 ; <+883 xe8b ; =0xe6 5 70x748 ; =0x74 8 ; _dyld sp, #0x10] sp], #0x20	3::_dyld_get_image_name(unsi; 95 48 d_func_lookup	ned int)	
 28 <b>D</b> Auto 2 (3)	* ± ± •	کہ 25 ج	Aweme )	Thread 1 ) 0 _dyld_get_ (11db) po \$arg1 <nil> (11db) p \$arg1 (unsigned long) \$2 = (11db) All Output 0</nil>	image_name Choose stack frame (hold Comman Ø	Line: 19 Col: 43 d to show full backtrace)	

然后试试:

Command + 点击

### 可以出现和前面一样的,完整的,函数调用堆栈:

	28	
		0_dyld_get_image_name
		1_CFGetHandleForLoadedLibrary
		• 2CFInitialize
		m 3 ImageLoaderMachO::dolmageInit(ImageLoader::LinkContext const&)
		Im 4 ImageLoaderMachO::doInitialization(ImageLoader::LinkContext const&)
		in 5 ImageLoader::recursiveInitialization(ImageLoader::LinkContext const&, unsiconst*, ImageLoader::InitializerTimingList&, ImageLoader::UninitedUpwards&)
		间 6 ImageLoader::recursiveInitialization(ImageLoader::LinkContext const&, unsiconst*, ImageLoader::InitializerTimingList&, ImageLoader::UninitedUpwards&)
		m 7 ImageLoader::processInitializers(ImageLoader::LinkContext const&, unsigned int, ImageLoader::InitializerTimingList&, ImageLoader::UninitedUpwards&)
		m 8 ImageLoader::runInitializers(ImageLoader::LinkContext const&, ImageLoader::InitializerTimingList&)
		9 dyld::initializeMainExecutable()
		10 dyld::_main(macho_header const*, unsigned long, int, char const**, char const**, unsigned long*)
		in 11 dyldbootstrap::start(dyld3::MachOLoaded const*, int, char const**, dyld3::MachOLoaded const*, unsigned long*)
1	Auto 🗘 🛛 💿	in 12_dyld_start

-》然后也知道了:

- 函数调用堆栈
  - o 英文专业叫法: backtrace
    - 此处是 完整的backtrace ,所以叫: full backtrace
    - 所以LLDB调试 函数调用堆栈 缩写是:
      - bt = backtrace

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 14:42:35

# 断点触发详情

#### Xcode触发断点时 还带提示第几次触发

举例:



断点触发到 \_\_\_11db\_unnamed\_symbol1653302\$\$AwemeCore 时,显示的:

Thread 1: breakpoint 25.1 (1)

表示:

- 所属线程: Thread 1
- 代码停下原因: 断点breakpoint
- 具体是哪个断点: 25.1
  - · 切换到断点列表中可以看到是:

	P Aweme master	Aw
<ul> <li>□ X T Q A ◇ A<sup>S</sup> ■ E</li> <li>≥ _logos_runction\$_ungroupea\$_ayia_get_image_vmaaar_siiae(unsi</li> <li>≥ _logos_function\$_ungrouped\$_dyld_get_image_vmaddr_slide(unsi</li> <li>≥ _logos_orig\$_ungrouped\$_dyld_get_image_vmaddr_slide</li> </ul>	Image: Second state   Image: Second state     Image:	ne.x 0 0_ur (+0)
<ul> <li>2 _logos_function\$_ungrouped\$_dyld_get_image_name</li> <li>2 _logos_orig\$_ungrouped\$_dyld_get_image_name</li> <li>2 +[_RxAnnotationInlineLoader load]</li> <li>2 _logos_function\$_ungrouped\$_dyld_image_count</li> <li>2lldb_unnamed_symbol148\$\$AwemeCore</li> <li>2lldb_unnamed_symbol1860893\$\$AwemeCore</li> <li>2lldb_unnamed_symbol1860893\$\$AwemeCore</li> <li>2dyld_register_func_for_add_image</li> <li>2 dyld3::_dyld_register_func_for_add_image(void (*)(mach_header c)</li> <li>2 _dyld_register_func_for_add_image libdyld.dylib</li> <li>2 systcl</li> <li>2 _logos_function\$_ungrouped\$sysctl</li> <li>3 getsectbynamefromheader_64</li> <li>2 +[AWECloudJailBreakUtility load]</li> <li>2 _logos_meta_method\$_ungrouped\$_RxAnnotationInlineLoader\$load</li> <li>2 _logos_function\$_ungrouped\$faccessat</li> <li>2lldb_unnamed_symbol2043\$\$AwemeCore</li> <li>3lldb_unnamed_symbol2043\$\$AwemeCore</li> </ul>	2 -> 0x114dc2e14 < 3 0x114dc2e18 < 4 0x114dc2e1c < 5 0x114dc2e20 < 6 0x114dc2e20 < 6 0x114dc2e20 < 7 0x114dc2e20 < 8 0x114dc2e20 < 9 0x114dc2e20 < 9 0x114dc2e30 < 10 0x114dc2e30 < 11 0x114dc2e30 < 12 0x114dc2e30 < 13 0x114dc2e30 < 13 0x114dc2e40 < 14 0x114dc2e40 < 15 0x114dc2e40 < 16 0x114dc2e40 < 17 0x114dc2e40 < 18 0x114dc2e50 < 18 0x114dc2e50 < 18 0x114dc2e50 < 18 0x114dc2e50 < 20 0x114dc2e50 < 20 0x114dc2e50 < 21 0x114dc2e50 < 22 0x114dc2e60 < 22 0x114dc2e60 < 23 0x114dc2e60 < 23 0x114dc2e68 < 3 0x114dc2e68 <	++62 ++42 ++12 ++12 ++22 ++22 ++32 ++32 ++32 ++42 ++42 ++4
<ul> <li>[AWELazyRegisterHandler handlerPointer]</li> <li>-[AWEUserRecommendFollowButton</li> <li>-[AWEUserRecommendFollowButton</li> <li>-[AWEUserRecommendFollowButton</li> <li>-[AWEUserRecommendMutiTagsView</li> <li>lldb_unnamed_symbol1674948\$\$</li> <li>Swift Error ID 15.1</li> <li>All Objective-C Exceptions ID 16.1</li> <li>-[AWEUIListTableViewCell actionBtnC</li> <li>-[AWEInviteSearchTableViewCell action</li> <li>Ignore 0 times before stopping</li> <li>Action Add Action</li> <li>Options Automatically continue after even</li> <li>lldb_unnamed_symbol1653302\$\$AwemeCore ID 25.1</li> <li>[AWESearchUser shouldFollowWhenTappedFollowButton] ID 28</li> <li>-[AWEUserModel shouldFollowWhenTappedFollowButton] ID 29</li> </ul>	24 0X1140C2e0C < 25 0X1140C2e070 < hbers or contain any white space. 2\$\$AwemeCore aluating actions	+92
<ul> <li>+ (● Filter</li> <li>■ 对应着是自己加的符号断点</li> </ul>	Auto ≎ ( )	

Ildb\_unnamed\_symbol1653302\$\$AwemeCore

• (1): 表示此处断点触发了第一次

• 后续如果再次触发,数字会依次增加,比如 (2)、(3)等等

Image: Constraint of the second sec	•••		P Aweme	Aweme	e 👌 🚺 iPho	me7_1331		Running Aweme on iPhone7_1331	<b>▲</b> +	
✓ ☑ Aweme PID 1321         ◎ ⑥           ☑ CPU         0%           ☑ Meme > ⑥ Thread 1) ৣ 0lidb_unnamed_symbol1653302\$\$\$AwemeCore:           ☑ Aveme > ⑥ Thread 1) ৣ 0lidb_unnamed_symbol1653302\$\$\$AwemeCore:           ☑ Aveme > ⑥ Thread 1) ৣ 0lidb_unnamed_symbol1653302\$\$AwemeCore:           ☑ Aveme > ⑥ Thread 1) ৣ 0lidb_unnamed_symbol1653302\$\$AwemeCore:           ☑ Aveme > ⑥ Thread 1) ৣ 0lidb_unnamed_symbol1653302\$\$AwemeCore:           ☑ > 0%         0%			$\mathbb{B} \mid < >$	hook_aweme.xm		wemeDylib.xm	0(db	unnamed_sy1653302\$\$AwemeCore	🗅 No Editor 🖹 🗍	Ŧ
CPU         0%         1         AkeneCore`lldb_unnamed_symbol1653302\$\$AkeneCore:         1           2         >>         X11462248         <+0>:         stp:         x2, x21, [sp: #=0x30]1         Image: Thread 1: breakpoint 25.1(2)	V Aweme PID 1321	0	Aweme >	🕕 Thread 1 👌 🔜 0lk	lb_unname	d_symbol1653302\$\$A	AwemeCore		<	▲ >
L II III III IIII IIII IIIIIIIIIIIIIII	CPU	0%	1 Awen	eCore`lldb_unna	med_symb	ol1653302\$\$Awem	eCore:			1
			2 ->	0x114dc2e14 <+0>:	stp	x22, x21, [sp,	#-0x30]!	= Threa	d 1: breakpoint 25.1 (2)	
E Memory Disabled 0 011140/2010 (+4): stp x20, x17, [5], #07.00]	Memory	Disabled	3	0x114dc2e18 <+4>:	stp	x20, x19, [sp,	#0x10]			· II
5 0x11402210 (10): add x29, 50, 602, 60, 100, 100, 100, 100, 100, 100, 100,	4 Frank Innert	Lilah	5	0x114dc2e20 <+12>:	add	x29, sp, #0x20	#0X20J	: =0x20		- 11
Ferrgy impact High 6 0x114dc2e24 <+165: mov x19, x8	Energy Impact	High	6	0x114dc2e24 <+16>:	mov	x19, x0		1 - 0.420		- 11
□ Disk Zero KB/s 7 0x114dc2e28 <+20>: add x0, x0, #0x30 ; =0x30	E Disk	Zero KB/s	7	0x114dc2e28 <+20>:	add	x0, x0, #0x30		; =0x30		- 11
8 0x114dc2e2c <+24>: bl 0x1161f98d8		1	8	0x114dc2e2c <+24>:	bl	0x1161f98d8				- 11
	🛞 Network	Zero KB/s	9	0x114dc2e30 <+28>:	mov	x20, x0	01			- 11
10 0x1140C2034 +432>: 101 X0, (X1Y, #0X20]	Thread 1 Queue, cam apple main thread (agric)		10	0x1140c2e34 <+32>: 0x114dc2e38 <+36>:	adro	x0, [x19, #0x20	01			- 11
	Thread T Quede: com.apple.main-thread (senar)		12	0x114dc2e3c <+40>:	ldr	x1, [x8, #0x876	01			- 11
13 0x114dc2e40 <+44>: bl 0x114df98fc	UIIdb_unnamed_symbol1653302\$\$AwemeCore		13	0x114dc2e40 <+44>:	bl	0x1161f98fc				- 11
□ 1lldb_unnamed_symbol1023498\$\$AwemeCore 14 0x114dc2e44 <+48>: bl 0x114dc3ff4 ;lldb_unnamed_symbol1653325\$\$AwemeCore	1Idb_unnamed_symbol1023498\$\$AwemeCore		14	0x114dc2e44 <+48>:	bl	0x114dc3ff4		;lldb_unnamed_symbol1	553325\$\$AwemeCore	- 11
2logos_method\$_ungrouped\$AWEUIListTableViewCell\$actionBtnClick 15 0x114dc2e48 <+52>: mov x21, x0	2 _logos_method\$_ungrouped\$AWEUIListTableViewCell\$action	nBtnClick	15	0x114dc2e48 <+52>:	mov	x21, x0				- 11
I 3 _logos_method\$_ungrouped\$AWEInviteSearchTableViewCell\$actionB          16       0x114dc2e4c       x6       x8       -56809         10       0x114dc2e4c       x6       x8       -56809	3 _logos_method\$_ungrouped\$AWEInviteSearchTableViewCell	\$actionB	16	0x114dc2e4c <+56>:	adrp	x8, -50800	01			- 11
4 -[UlApplication sendAction:to:from:forEvent:]	4 -[UIApplication sendAction:to:from:forEvent:]		17	0x114dc2e50 <+00>: 0x114dc2e56 <+60>:	10r	AV1161f98fc	01			
32 start 19 0x114d22658 <<68>; moy x22, x8	32 start		19	0x114dc2e58 <+68>:	mov	x22, x0				
Thread 4 Queue: com heimdallr.runloop.observer (serial) 20 0x114dc2e5c <+72>: bl 0x114dc400c ;lldb_unnamed_symbol1653328\$\$AwemeCore	> Thread 4 Queue: com.heimdallr.runloop.observer (serial)		20	0x114dc2e5c <+72>:	bl	0x114dc400c		;lldb_unnamed_symbol1e	553328\$\$AwemeCore	
21 Øx114dc2e60 <+76>: adrp x8, -50702	> Thread 5		21	0x114dc2e60 <+76>:	adrp	x8, -50702				
22 0x114d22e64 <80>: add x8, x8, #0x1a8 ; =0x1a8	Thread 6		22	0x114dc2e64 <+80>:	add	x8, x8, #0x1a8		; =0x1a8		
23 0X11402060 4484>: 10p X2, X3, [X19, #0X20]			23	0x114dc2e68 <+84>:	Idp	x2, x3, [x19, 4	#0x20]			
JavaScriptCore omalios scavenger (11) 22 6x14402260 (+00): dulp 47, 50702 := ex1a0	> U JavaScriptCore bmalloc scavenger (11)		24	0x114dc2e70 <+92>:	add	x9, x9, #0x1a0		: =0x1a0		
26 0x114dc2e74 <+96>: cmp w22, #0x0 ; =0x0	WebThread (12)		26	0x114dc2e74 <+96>:	cmp	w22, #0x0		; =0×0		
Com.apple.uikit.eventfetch-thread (13) 27 0x114dc2e78 <+100>: csel x8, x8, x9, ne	> () com.apple.uikit.eventfetch-thread (13)		27	0x114dc2e78 <+100>	: csel	x8, x8, x9, ne				
> O ThreadPoolServiceThread (14)	> () ThreadPoolServiceThread (14)			V V T R S	0 4			ldb uppamed symbol1653302\$\$Awer	meCore Line: 2 Col: 1	
> O ThreadPoolBackgroundWorker (16)	> () ThreadPoolBackgroundWorker (16)				G /				liecore Line. 2 col. 1	
> O NetworkConfigWatcher (17) hook_AwemeDylib.xm TTMacroManagerSisDebug:	> () NetworkConfigWatcher (17)						ho	ook_ AwemeDylib.xm TTMacroMa	inager\$isDebug:	
> () NetworkCacheThread-0-tid:27395 (18) 2022-04-28 10:15:08.283753+0800 Aweme[1321:22075]	NetworkCacheThread-0-tid:27395 (18)						2022-0	04-28 10:15:08.283753+0800	weme[1321:22075]	
> NetworkCacheThread-1-tid:37379 (19)	> () NetworkCacheThread-1-tid:37379 (19)						ho	ook_ AwemeDylib.xm		
Chrome File Thread (20)	> () Chrome File Thread (20)						2022-6	VEINVITESearchTableviewCell: 04-28 10:15:10.978514+0800 /	weme[1321:22075]	
Chrome Network (O Thread (21) hook_ AwemeDylib.xm	Chrome Network IO Thread (21)						ho	ook_ AwemeDylib.xm		
AMEULLIStTableviewCellSactionEtholicked:	Thread 23 Queue: com apple coremedia d. onsession clientcallba	ack (serial)					AV	VEUIListTableViewCell\$action	BtnClicked:	
2022-04-20 10:11:40-40-00-0000 Awene(1321:220/3) hook	> volouid mdi file, manager (30)	lon (bornar)					2022-0	ook_ AwemeDylib.xm	Aweme[1321:220/5]	
AWEInviteSearchTableViewCell\$rightButtonClickedBl	Com video mdl network cession (21)						AV	EInviteSearchTableViewCells	FrightButtonClicked	iB1
C C C C C C C C C C C C C C C C C C C							2022-0	ck: retBlock= <nsmallocbloc A4_28 10:15:13 803372:0800 /</nsmallocbloc 	k: 0x287e3f9c0>	
hok_averepylib.xm	hequesineceiver (32)						h	ook_ AwemeDylib.xm		
AMEInviteSearchTableViewCell\$rightButtonClickedBl	MediaLoad (33)						AV	EInviteSearchTableViewCells	FrightButtonClicked	iB1
> U MediaLoad (34)	> MediaLoad (34)						(11db)	ck: retBlock= <nsmallocbloc< th=""><th>K: 0x287e3f9c0&gt;</th><th></th></nsmallocbloc<>	K: 0x287e3f9c0>	
	P Filter		Auto 0		() F		All Out	put ¢ 🕞 Filter	ê (	וחו

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 11:48:18

## 停下来的原因

Xcode代码停下来, 会显示具体原因

用Xcode去调试程序时:

代码有时候会停下来 暂定运行

此时可以看出对应的代码停下来的原因:

• SIGTRAP: 进入了某个trap 陷阱? 触发了某个条件而进入了陷阱?

0

• breakpoint = 之前自己设置的某个断点,生效了,触发了某个断点,导致代码停下来

о

• SIGABRT=abort终止 = 发生了某些异常导致程序终止

。 ● 异常:

### • EXC\_RESOURCE RESOURCE\_TYPE\_MEMORY

- Thread 149: EXC\_RESOURCE RESOURCE\_TYPE\_MEMORY (limit=1850 MB, unused=0x0)
  - 🔯 Aweme > 📔 iPhoneX\_137 Running Aweme on iPhoneX\_137 🔺 11 🕂 ••• 🔳 🕨 🚨 Aweme BB | < > 🕆 hook\_aweme.xm 🚺 0 mach\_msg\_trap 🖓 hook\_dylib.xm 🖗 AntiAntiDebug.m 🖓 hook\_misc.xm 🖗 AwemeD 🚉 🕞 Aweme ) () Thread 149 ) 🖬 0 mach\_msg\_trap Invest 149 ) □ 0 mach\_msg\_trap
     Ibsystem\_kernel.dylib`mach\_msg\_trap
     Ox19227d198 <+0: mov x16, #=0x1f
     0x19227d198 <+0: svc #0x88
     4 → 0x19227d198 <+8: ret
     5 < A > Aweme PID 46705

     Image: Aweme PID 46705
     CPU 13% Disabled Thread 149: EXC\_RESOURCE RESOURCE\_TYPE\_MEMORY (limit=1850 MB, unused=0x0) forsy impact very High
    the Himself - a debug - also - Source
     O Thread 5
     O Thread 6
     O JavaScriptCore bmalloc scavenger (9)
     O WebThread (10) 
     >
     WebThread (10)

     >
     0 com.apple.uikit.eventfetch-thread (11)

     >
     0 NetworkConfigWatcher (12)

     >
     0 NetworkCacheThread-0-tid:38659 (13)

     >
     0 NetworkCacheThread-1-tid:38403 (14)
     OrderworkCacherinead-Fata.se405 (14)
     O com.hmd.mach\_server (15)
     O com.hmd.crash.listener (16)
     O com.apple.CoreMotion.MotionThread (18) ■ ID > ↓ 1 0 > 8 7 Memme) Thread 14 0 OwnetUnsg\_trap Q. Find v md mex\_providerstart e: 5 Col: 1 
     ws.j.pvv.sigs/v
     2 0\_c\_connect\_method

     2022-64-92 15:16:1
     3 10ConnectCallMethod

     2022-64-92 15:17:1
     4 10SufraceClentLookupFromMachPort

     2022-64-92 15:17:1
     5 0 Glorance initWinMachPort

     2022-64-92 15:17:1
     6 Figheroits\_CreatePixeBufferFormEniltEndAtionOxta

     TMacroManager
     2 0 Glorance initWinMachPort

     202-64-92 15:17:0
     9 Glogatch\_call.bock\_main\_messa

     etvice is dis informed
     9 Glogatch\_call.bock\_main\_messa

     202-64-92 15:17:0
     9 Glogatch\_lans\_erial\_crain

     202-64-92 15:17:0
     10 dispatch\_lans\_invoke

     TMacroManager
     12 \_clinpatch\_lans\_invoke

     TMacroManager
     12 \_clinpatch\_lans\_wrisd\_drain

     2022-64-92 15:17:0
     11 \_clinpatch\_lans\_invoke

     TMacroManager
     12 \_clinpatch\_lans\_drain

     2022-64-92 15:17:0
     12 \_clinpatch\_lans\_drain

     2022-64-92 15:17:0
     13 \_pthread\_wqthread

     AMFEl ptyrus 1
     13 \_pthread\_wqthread
     > Done > 🕕 Thread 19 > 
     ThreadPoolServiceThread (20) ThreadPoolServiceThead (20)
     ThreadPoolBackgroundWorker (22)
     Chrome File Thread (23)
     Chrome Network IO Thread (24) O Thread 25
     Thread 25
     Thread 35 Queue: com.heimdal...cop.observer (serial)
     Thread 35 Queue: com.apple.c...clientcallback (serial) > () vclould.mdl.file\_manager (38) Velocitization\_interpretations
     Com.video.mdl.network\_session (39)
     RequestReceiver (40)
     MediaLoad (41) 🗑 Filter Auto 0 () 🐨 Filter All Output 0 1 00
  - Thread 1: EXC\_RESOURCE RESOURCE\_TYPE\_MEMORY (limit=1450 MB, unused=0x0)



![](_page_27_Figure_2.jpeg)

总之:

Xcode调试期间,代码停下来,一般会显示对应的原因的。

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 11:53:59

## 箭头指向的指令

0x1090ff40c <+100>: bl

0x110cfbfe4

当调试时,代码停下来时,会看到的汇编代码中的有个: 箭头 = ->

而 -> 指向的是将要运行的指令代码,不是当前正在运行的代码

举例:

•••	Aweme 🔝 Aweme ) 🗓 iPhone7_1331 Running Aweme on iPhone7_1331	· 🔺 + 🖪
	Bit     Image: Second symbol       Bit     Image: Second symbol <td>≣0   [+</td>	≣0   [+
Aveme PID 24101     General Aveme PID 24101     General Aveme PID 24101     General Aveme PID 24101     General Aveme Ave		(1)     (2)     (
0x1090ff404 <<92 : add x0, x0, #0xbc 0x1090ff408 <<96 : bl 0x110cfbfe8 0x1090ff40c <<100 : bl 0x110cfbfe4 > 0x1090ff410 <<104 : cbz w0, 0x1090ff448	<pre>; =0xbc ;lldb_unnamed_symbol1205873\$\$AwemeCore ;lldb_unnamed_symbol1205872\$\$AwemeCore ; &lt;+160&gt;</pre>	
当前 -> 指向的:		
-> 0x1090ff410 <+104>: cbz w0, 0x1090ff448	; <+160>	
是 将要运行的 下一行的代码=指令 而当前正在运行的指令,对应着箭头 -> 的上面的	5—行:	

crifan.org,使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 14:27:58

; \_\_\_lldb\_unnamed\_symbol1205872\$\$AwemeCore

## 切换进程视图

#### 在不同视图之间切换:

#### 在调试期间,新点击了:

- Debug Memory Graph
- Debug View Hierarchy
- 后,回不去调试的Thread了

#### 后来发现了,是点击:

#### View process in different ways

![](_page_29_Picture_9.jpeg)

即可看到几种方式:

• 当前的是: View UI Hierarchy

0

切换到:

• View Process by Thread

o

就是Debug默认的,常见的形式了:

以线程方式查看进程,其中能看到函数调用堆栈的内容:

•••		weme	🛛 Aweme 🔪 📋	iPhone7P_1341	Running Aweme on iPhone7F	P_1341 🔒 14	+	
🗀 🛛 🎞 🔍 🛆 🗇 🗗 🗐	88	< >   I View hierarch	y for Aweme	🔟 0 _dyld_get_image_name			≡0	
X SR Q A O O	Image: Constraint of the second sec	View hierardt verene ) ● Thread 1 ) ● 0 liddy1d. dy1ib '_dy1 → 0x1ad82551a <+ 0x1ad825524 <+ 0x1ad825524 <+ 0x1ad825524 <+ 0x1ad825524 <+ 0x1ad825524 <+ 0x1ad825538 <+ 0x1ad825538 <+ 0x1ad825538 <+ 0x1ad825534 <+ 0x1ad825534 <+ 0x1ad825534 <+ 0x1ad825544 <+ 0x1ad825554 <+ 0x1ad825557 <+ 0x1ad825577 <+ 0x1ad825578	y for Aweme dyld_get_imag det_imag(det_imag) >: stp >: stp >: add 12>: mov 16>: adTp 22>: add 12>: dt 22>: add 12>: dt 22>: ld 1d 55>: cbz 23>: mov 36>: ldp 469: ldp 469: adTp 72>: add 56>: cbrz 569: adTp 72>: add 769: ldp 969: ldp 189: br	<pre></pre>	<pre>; =0x10 ; =0x698 ; &lt;+48&gt; ; dy1d3::_dy1d_get_image_nam ; &lt;+88&gt; ; =0xe0b ; =0x748 ; _dy1d_func_lookup</pre>	Thread 1: instruction	EC.	r
	28		~ D ~	1				

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 14:33:29

### **Report Navigator**

## 对于项目的编译和运行的详细过程,可以通过Report navigator中查 看

Xcode -> Show the Report navigator -> 点击对应 Build 或 Run ->右边即可显示出编译或运行的详细过程log日志

• Build

#### 。 举例

![](_page_31_Picture_6.jpeg)

o 导出: 点击 Export 导出 txt 格式的日志

![](_page_31_Figure_8.jpeg)

• Run

![](_page_31_Figure_10.jpeg)

	■ 28 < > × 2 quic	k-resign.sh 🕒 Run Aweme	🗅 Run Aweme 🕒 Run Aweme	E hook_dylib.xm E hook_misc.xm	AwemeD
Local Cloud	Run Aweme : 2:37:03 PM				
Du Casura Du Tinta	OD]C[3/981]: CIASS	MILKeversiblevalueiransto	rmer is implemented in both /pri	vare/var/containers/sundie/Application	n/4E84/8F9-
By Group By Time	B026-4C5B-89F7-AD9 Application/4E8478	51806BB99/Aweme.app/Framew F9-B026-4C5B-89F7-AD951806	orks/byteaudio.framework/byteaud BB99/Aweme.app/Frameworks/AwemeC	io (0x1168ec878) and /private/var/cont pre.framework/AwemeCore (0x103a63a00).	. One of the t
> Run Aweme 2022/3/25, 6:42 PM	be used. Which one	is undefined.	2402452] [PDI] NHI Americana CODE	4 VED.1 CLD.0-102414000	
Build Aweme 2022/3/25, 6:42 PM	A 2022-03-25 18:43:50	0.740730+0000 Aweme[37901:	2493652] [BDL] NM: Awemecore, CORE	10, VER:1, SLD:0X102014000	
▷ Run Aweme 2022/3/25, 6:41 PM	2022-03-25 18:43:5	6.747272+0800 Aweme[37981: 007983s.BD:0.001718s(29485	2493652] [BDL] ).EX 0C:0.004198s.EX CE:0.005064	s. FX SFI (lock): 0.063331s	
Build Aweme 2022/3/25, 6:41 PM	<u>A</u>				
Run Aweme 2022/3/25, 6:38 PM	2022-03-25 18:43:5	6.747294+0800 Aweme[37981:	2493652] [BDL] A11:0.126278s,A11	-FX_SEL:0.062947s	
Build Aweme 2022/3/25, 6:38 PM	A espresso version:	2.7.11.0520			
▷ Run Aweme 2022/3/25, 6:36 PM	mobilecv2: 1.9.0.1	913			
Build Aweme 2022/3/25, 6:36 PM	🔺 🐂 🎉	!! congratulations!!! 🎉			
▷ Run Aweme 2022/3/25, 6:34 PM		insert dylib success			
Build Aweme 2022/3/25, 6:34 PM	A 2022-03-25 18:43:5 2022-03-25 18:43:5	7.414357+0800 Aweme[37981: 7.442306+0800 Aweme[37981:	2493052] [AntiAntiDebug Init] 2493652] hook_ AwemeDylib.xm _lo	gosLocalCtor_46ba9f2a: AwemeDylib cto	ar
Clean Aweme 2022/3/25, 6:34 PM	[MethodTrace]			_	
Run Aweme 2022/3/25, 6:33 PM	ttns://aithub.com	OCMethodTrace(Usage) /omycodec/OCMethodTrace/bl	ob/mastor/DEADME md		
Build Aweme 2022/3/25, 6:32 PM	A &	OCMethodTrace(Usage)			
Clean Aweme 2022/3/25, 6:32 PM	[MethodTrace] logL	evel: 0: logWhen: 0 traceF	lag: 2 traceObject: 0(未指定类)		
▷ Run Aweme 2022/3/25, 6:30 PM	[MethodTrace] Metho 2022-03-25 18:43:5	od Trace is disabled 7.477868+8888 Aweme[37981*]	2493652] You've implemented =[<]	[AnnlicationDelegate>	
Build Aweme 2022/3/25, 6:30 PM	A application:didRec	eiveRemoteNotification:fet	chCompletionHandler:], but you s	till need to add "remote-notification"	" to the list
Run Aweme 2022/3/25, 6:28 PM	Supported UIBackgr 2022-03-25 18:43:5	oundModes in your Info.pli 7.619404+0800 Aweme[37981::	st. 2493874] *** Terminating app due	to uncaught exception 'NSInvalidArgum	mentException
Build Aweme 2022/3/25, 6:28 PM	A '*** -[NSCFConst	antString stringByAppending	gString:]: nil argument'		
▷ Run Aweme 2022/3/25, 6:26 PM	(0x1924a9654 0x192	1cbbcc 0x192771fbc 0x1086a	a6c0 0x1086aa608 0x1086aa480 0x1	086aa234 0x11282283c 0x1026f7730 0x102	26f9044 0x108
Build Aweme 2022/3/25, 6:26 PM	A 0x1086e4930 0x10f6 0x113b29e2c 0x110c	81d24 0x10f67d8d4 0x10f67d	874 0x1026f7730 0x1026f9044 0x10 744 0x108d59f04 0x108d3faec 0x10	86ebf34 0x10f6b16d8 0x111021adc 0x1086 26f6338 0x1026f7730 0x1026fe760 0x1026	6963a0 0x1086
Clean Aweme 2022/3/25, 6:26 PM	0x1921c0b74 0x1921	c3740)			011200 012027
Clean Aweme 2022/3/25, 6:13 PM	libc++abi.dylib: to terminating with u	erminating with uncaught e neaught exception of type	xception of type NSException		
Run Aweme 2022/3/25, 6:03 PM	*** Terminating ap	p due to uncaught exception	n 'NSInvalidArgumentException',	reason: '*** -[NSCFConstantString	
Build Aweme 2022/3/25, 6:03 PM	A StringByAppendingS Message from debug	tring:j: nii argument' ger: failed to send the k i	packet		
Clean Aweme 2022/3/25, 6:03 PM		- 11.174 - 1899 - 1899 - 1899 - 1899 - 1899 - 1899 - 1899 - 1899 - 1899 - 1899 - 1899 - 1899 - 1899 - 1899 - 18			
Clean Aweme 2022/3/25, 5:56 PM	-				
Run Aweme 2022/3/25, 5:44 PM			exception NSINVAILOArgument	Corperate	cancouring
Build Aweme 2022/3/25, 5:44 PM	<u>A</u>		<pre>stringsympteringstring:j: n w** First throw call stack:</pre>	11 argument	
Run Aweme 2022/3/25, 5:41 PM			(0x1924a9654 0x1921cbbcc 0x19277	lfbc 0x1086aa6c0 0x1086aa608 0x1086aa4	480 0x1086aa23
Build Aweme 2022/3/25, 5:41 PM	<u>A</u>		0x11282283c 0x1026f7730 0x10	26f9044 0x1086aa1dc 0x1086e4930 0x10f6	681d24 0x10f67
Run Aweme 2022/3/25, 5:38 PM			0x1076/08/4 0x10267/730 0x10 0x10869c218 0x113b29e2c 0x11	019044 0x1080eDT34 0x10ToD16d8 0x1110 0c62fdc 0x108c9ed68 0x108d597d4 0x108d	#2140C 0X10869 d59f04 0x108d3
Build Aweme 2022/3/25, 5:38 PM	<b>A</b>		0x1026f6338 0x1026f7730 0x10	26fe740 0x1026ff2e0 0x10270a6c4 0x1921	1c0b74 0x19210
Class Aurama 2002/2/25 5:27 DM		Q Filter	lihottahi dulih: tarminating with	unraught evention of tune NCEventi	ion
N/7 1 T 1 T 1 T 1 T 1 T 1 T 1 T 1 T 1 T 1	AULO CO (1)	O FIILUI	Mil Output v	Filter	

TODO:

【已解决】XCode项目的编译链接安装等内部详细日志和过程

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 14:15:49

### **Build Phases**

### Run Script

#### Run Script对应着Build中的Script-xxx.sh

Xcode -> Target -> Build Phases -> Run Script

#### 其中输入的要运行的脚本代码

/opt/MonkeyDev/Tools/pack.sh

	🛃 Aweme		Aweme > iPhoneX_137	Finished running Aweme on iPhoneX_137	+ 📑				
	BB < > Aweme.xco	deproj			≓ ⊡				
V 🔼 Aweme	🚨 Aweme				< 🔺 >				
~ 🗃 Aweme		General	Signing & Capabilities Resource Tags	Info Build Settings Build Phases Build Rules					
🖾 icon.png	PROJECT -	+		() Filter					
Info.plist	🛃 Aweme			Add output file list files here					
auick-resign.sh	TARGETS		+ -						
> Config	()) Aweme								
✓	m AwemeDylib	> Copy Bundle	e Resources (2 items)		×				
I 抖音 Aweme_18.9.0_svc0x80ToNop.ipa Ⅰ put ipa or app here		> Copy Files (	(1 item)		×				
> 🚞 AwemeDylib		✓ Run Script			×				
> 🔚 Frameworks			Shell /hin/sh						
			1 (ant/MankayDay/Tanls/pask sh padasian						
			2	k.sn codesign					
			Run script: 🗌 For install builds only						
			Based on dependency anal	lysis					
			Will skip script in incremental						
			Show environment variables in build log						
			Input Files		1				
				Add input files here					
			+ -						
			Input File Lists						
	+ - 🕞 Filter			Add input file list files here					
	_								

#### ->最终对应着:

#### Build日志中的:

- Script-9873AEB427EDB873002EA2A0.sh
  - o 完整路径: /Users/crifan/Library/Developer/Xcode/DerivedData/Awemefswcidjoxbkibsdwekuzlsfcdqls/Build/Intermediates.noindex/Aweme.build/Debug-iphoneos/Aweme.build/Script-9873AEB427EDB873002EA2A0.sh
  - o 脚本内容是:

#!/bin/sh
/opt/MonkeyDev/Tools/pack.sh

o 对应的运行脚本的命令是

PhaseScriptExecution Run\ Script /Users/crifan/Library/Developer/Xcode/DerivedData/Aweme-fswcidjoxbkibsdwekuzlsfcdqls/ Build/Intermediates.noindex/Aweme.build/Debug-iphoneos/Aweme.build/Script-9873AEB427EDB873002EA2A0.sh (in target 'Awem e' from project 'Aweme')

```
______/Users/crifan/dev/DevRoot/Aweme/MonkeyDev/Aweme_18.9.0
```

/bin/sh -c /Users/crifan/Library/Developer/Xcode/DerivedData/Aweme-fswcidjoxbkibsdwekuzlsfcdqls/Build/Intermediate s.noindex/Aweme.build/Debug-iphoneos/Aweme.build/Script-9873AEB427EDB873002EA2A0.sh

8 = 9 1	•	🔼 Aweme	Aweme >	iPhoneX_137	Finished running Aweme on iPhoneX_137	- 🖪
		😤   < >   🔄 quick-resign.st	n 🕒 Run Aweme	Build Aweme - Log	🗅 Run Aweme 📄 hook_dylib.xm 📄 hook_mis	
Local Cloud		Build Aweme				< 🔺 >
By Group By Time		All Recent All Message	All Issues Errors O	nly	Export 🕲 Filter	
By Grau By Ima     By Grau By Ima     Build Aweme 2022/3/25, 6:42 PM     Build Aweme 2022/3/25, 6:42 PM     Build Aweme 2022/3/25, 6:41 PM     Build Aweme 2022/3/25, 6:38 PM     Build Aweme 2022/3/25, 6:34 PM     Build Aweme 2022/3/25, 6:33 PM     Build Aweme 2022/3/25, 6:32 PM     Glean Aweme 202/3/25, 6:32 PM     Glean Aweme 202/3/25, 6:32 PM     Glean Awe		All         Recent         All Message           ●         Write Awares-roject-team         Write Awares-roject-team           ●         Write Awares-roject-team         Write Awares-roject-team           ●         Write In-product-headers         ●           ●         Write In-product-headers         ●           ●         Write In-product-headers         ●           ●         Write In-product-headers         ●           ●         Write Script-B73AEB427         ●           ●         Write Awares-training mack()         Headers-training mack()           ■         Basel Thrite Plattarts-training mack()         Headers-Thrite Plattarts-training mack()           ■ </td <td>All seuse Error D All seuse Error D haddrs.hmap 0.1 seconds econds DBA2002R20.ab 0.1 EDB873002R20.ab 0.1 E</td> <td>nty seconds if an/Library/Developer/XX hep/br/Akemp_11, 30 at 34/ thorp br/Akemp_11, 30 at 34/ thorp br/Akemp_11, 30 at 34/ br/Akemp/Akemp_11, 30 at 34/ br/Akemp/Akemp_12, 30 at 34/ Developer/Xxodo/DerivedDa Jones/Dev/Akemp_13, 30 at 34/ Developer/Xxodo/DerivedDa Jones/Dev/Akemp_14, 30 at 34/ Akemp/Akemp/14, 34/ Akemp/Akemp/14, 34/ Akemp/Akemp/14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Stop verify appl appl appl appl appl Akemp-MonkeyDev/Akemp_13, 3 appl appl appl Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp_14, 34/</td> <td>Dde/DerivedData/Aweme-fswcidjoxkbibsdwekuzlsfcdqls/Build/ pt-B93AEB427EDB873092EA2A0.sh (in target 'Aweme' from project 'Aweme') eme-fswcidjoxkbibsdwekuzlsfcdqls/Build/Products/Debug-iphoneos a /B. 9.d/Aweme/Info.plist eme/Tap a /B. 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 110.50.d/aweme/TargetApp/Aweme.app 2.10.50.d/aweme/TargetApp/Aweme.app 2.10.50.d/aweme/TargetApp/Aweme.app 2.10.50.d/aweme/TargetApp/Aweme.app 2.10.50.d/aweme/TargetApp/Aweme.app</td> <td>ø,</td>	All seuse Error D All seuse Error D haddrs.hmap 0.1 seconds econds DBA2002R20.ab 0.1 EDB873002R20.ab 0.1 E	nty seconds if an/Library/Developer/XX hep/br/Akemp_11, 30 at 34/ thorp br/Akemp_11, 30 at 34/ thorp br/Akemp_11, 30 at 34/ br/Akemp/Akemp_11, 30 at 34/ br/Akemp/Akemp_12, 30 at 34/ Developer/Xxodo/DerivedDa Jones/Dev/Akemp_13, 30 at 34/ Developer/Xxodo/DerivedDa Jones/Dev/Akemp_14, 30 at 34/ Akemp/Akemp/14, 34/ Akemp/Akemp/14, 34/ Akemp/Akemp/14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Stop verify appl appl appl appl appl Akemp-MonkeyDev/Akemp_13, 3 appl appl appl Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp/Akemp_14, 34/ Akemp_14, 34/	Dde/DerivedData/Aweme-fswcidjoxkbibsdwekuzlsfcdqls/Build/ pt-B93AEB427EDB873092EA2A0.sh (in target 'Aweme' from project 'Aweme') eme-fswcidjoxkbibsdwekuzlsfcdqls/Build/Products/Debug-iphoneos a /B. 9.d/Aweme/Info.plist eme/Tap a /B. 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 9.d/Aweme/TargetApp/Aweme.app 110.50.d/aweme/TargetApp/Aweme.app 2.10.50.d/aweme/TargetApp/Aweme.app 2.10.50.d/aweme/TargetApp/Aweme.app 2.10.50.d/aweme/TargetApp/Aweme.app 2.10.50.d/aweme/TargetApp/Aweme.app	ø,
Clean Aweme 2022/3/25 5:56 PM			POPULATION PROPERTY IN THE POPULATION PROPERTY INTE POPULATION PROPERTY IN	A STREET AND A ST	III CRAMMAR INCOME ANA	

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 14:11:40

# 其他

# 可以通过打开文件,打开另外一个项目中的文件

之前调试抖音的Xcode项目,期间会触发断点运行到,另外一个插件的代码

所以之前是当前Xcode项目自动打开了另外项目中的文件:

![](_page_35_Picture_5.jpeg)

此处是希望继续保持的状态。

另外也担心不小心关闭掉了打开的文件,好像就没机会再去打开到对应代码文件了? 比如想要添加新的断点,然后就不方便再去重新找到另外项目中的代码文件,再去新增断点 后来突然想到:

其实可以通过Xcode中的,打开文件,的功能,去打开,任何位置,包括别的Xcode项目中的文件的

去试试,此处故意去打开一个,之前没有打开的文件:

hook\_init.xm

Ś	Xcode	File	Edit	View	Find	Navigate	Editor	Product [
• •	•	New				>		🛃 Aweme
	=	Add F	iles to	"Aweme	"…	∧ ₩ ブ		昭   < >
		Add P	ackage	es		hook_awem		
	Aweme	Open				¥0		/39 }
2	Cycript	Open	Recen	t		>		740
~	Aweme	Open	Quick			0.00		741 %end
	icon.	Open	QUICKI	y		1 40 0		743 /*===
	Info.	Close	Other	Tabs		\7₩W		744 Hook
;	Scrip	Close	"hook	_aweme	.xm"	^₩W		745 =====
`	🖌 🚞 Conf	Close	Other	Editors		へて <b>企</b> 業W		746
	I MI	Close	Other	Window	Tabs	℃☆#W		747 %hook
	Targ	Close	Projec	t		~~~ #W		749 - (vo
	► 抖	0.030				20011		750 i
	🖻 pu	Save	All			\7₩S		751 %
> =	Aweme	Save	As			℃企業 S		752 }
~	Frame	Rever	t to Sa	ved				754 //-(0
		Unloc	k					755 -(id)
	- Four	Expor	t					756 i
	- Four							757 i
	CIKIT	Show	in Find	ler				758 r
		Open	in Tab			くまつ		760
		Open	in New	Window	v			761 //-(C
		Open	with E	xternal E	ditor			
		Packa	iges			>		-
		Save Project	As Wor ct Setti	kspace ngs				
		Page	Setup			企業 P		
		Print.				жР		

Kcode File Edit View Find	Navigate Editor	Product Debug	Source Control	Window Help 🏾 🅭	🖌 🏤 1	5 🖾 Q O 🔀 🔶 🖻
•••	•	Aweme	(i) /	Aweme > iPhoneX_137		Finished running Aweme on iPhoneX_13
	Favorites	<> ••••	000 ~		٢	Q Search
<ul> <li>X 证 Q A 《 6%</li> <li>Cycript_scripts</li> <li>Cycript_scripts</li> <li>Gon.png</li> <li>Info.plist</li> <li>Scripts</li> <li>Config</li> <li>MDConfig.plist</li> <li>TargetApp</li> <li>拼音 Aweme_18.9.0_restoredSym</li> <li>put lipa or app here</li> <li>AwemeDylib</li> <li>Frameworks</li> <li>JavaScriptCore.framework</li> <li>Foundation.framework</li> <li>UlKit.framework</li> </ul>	Favorites ● dev ④ Downloads ▲ Applications ● crifan ④ Recents ■ Desktop ● Documents ● Macintosh ICloud ● ICloud Drive Media ↑ Music ④ Photos ■ Movies Tags ● 黄色 ● 续色 ● 灰色 ● 紫色 ● 灰色 ● 紫色 ● 紫色 ● 武色	C C C C C C C C C C C C C C C C C C C	> > >deproj >	Previous 30 Days m hook_machO.mm hook_machO.mm hook_machO.xm Ξ月 AlterSystem.h alterSystem.m hook_ctcarrier.mm hook_ctcarrier.mm hook_ctcarrier.xm m hook_sysctl.xm m hook_sysctl.xm m hook_sysctl.xm m hook_uidevice.mm hook_uidevice.mm hook_uidevice.xm —月 m hook_init.mm hook_init.mm hook_init.xm m hook_misc.rm hook_misc.xm 2021 i CommonConfig.h m hook_aweme.xm hook_dyld.mm hook_dyld.mm hook_dyld.mm hook_dyld.mm		Q Search       الله         Carlow       الله         book_init.xm       Information         Document - 5 KB       Information         Created       2022年1月24日下午3:39         Modified       2022年3月17日上午10:17         Last opened       2022/4/15上午9:24         Tags       Add Tags
	○ 个人 つ All Tags			hook_dylib.xm m hook_openFile_C.mm	11	р е  и t
					U7 77 U7 - UU	Cancel Open t

![](_page_37_Figure_2.jpeg)

•••	P master	â	> iPhoneX_137	Build Succeeded   Today	y at 4:47 PM 🔺 🕂	
	88   < >   E	control hook_init.xm	hook_aweme.xm	hook_dylib.xm	🗉 hook_misc.xm 🖨 🚞   💽	Ð
Buildtime (52) Runtime		· [	hook_init.xm > No Selection		< 🔺 >	>
<ul> <li>Control (CZ) Relation (CZ) Relation</li></ul>	I         /*           1         /*           2         File: h           3         Functio           4         Author:           5         */           6         *           7         #import           8         #import           10         #import           11         #import           12         '/*           13         /*=====           14         Const           15         ======           16         // const NS           18         //#defin           19         /*======           20         /*======           21         Goba           22         ======           23         // all m           26         bool cfg           27         /* =====           28         // sub m           30         bool cfg           31         32           32         // sub m           33         bool cfg           34         4           35         // sub m           36         bool cfg           37	hook_init.xm on: iOS tweak global ini : Crifan Li <os log.h=""> "CommonConfig.h" "CrifanLibiOS.h" "CrifanLib.h" st st st st st st st st st st</os>	t t prove_nnixim / No Selection t t prove_nnixim / No Selection t prove_nnixim / No Selection t prove_nnixim / No Selection prove_nnixim / No Selection	references, ces/ plist	*/ 	
() Filter	44 bool cfg	gHookEnable_misc = true;			19 characters	1

而此处,当前Xcode项目中,并没有打开。

那故意把另外的Xcode项目关闭后,再去试试

#### 会弹框显示,单独显示

![](_page_38_Picture_4.jpeg)

然后继续去试了试,发现也是可以单独加断点的:

![](_page_39_Picture_1.jpeg)

-》基本上满足了我们的要求:

Xcode的A项目中,打开B项目中的文件

用途是:

A项目中的Xcode部分断点,会触发B项目中的文件的代码生效,用于调试查看B项目源码运行情况

注: B项目编译期间,是保留了 symbol 的,所以是有机会源码调试的。

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 14:03:16

# 附录

下面列出相关参考资料。

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 10:29:43

## 参考资料

- •【已解决】Mac中F7快捷键被占用
- 【无需解决】Xcode调试log日志窗口不输出log日志了
- •
- Ildb常用命令与调试技巧 掘金 (juejin.cn)
- •

crifan.org, 使用署名4.0国际(CC BY 4.0)协议发布 all right reserved, powered by Gitbook最后更新: 2022-10-31 11:04:46