

PLAY FUZZING MACHINE

hunting iOS/Osx kernel vulnerability automatically and Smartly

Lilang Wu (@Lilang_Wu), Moony Li(@Flyic)

About Me

- ❖ Lilang Wu
 - ❖ 4 years security experience
 - ❖ macOS/iOS malware/vulnerability
 - ❖ Fuzzing project
 - ❖ Twitter: @Lilang_Wu
 - ❖ BH USA 2019, 2018, BH EU 2018, HITB, CodeBlue

- ❖ Moony Li
 - ❖ 10 years security
 - ❖ MacOS/Android/iOS vulnerability
 - ❖ Vul hunt and exploit, SandBox dev
 - ❖ Twitter: @Flyic
 - ❖ BH 2019,2018,2016...



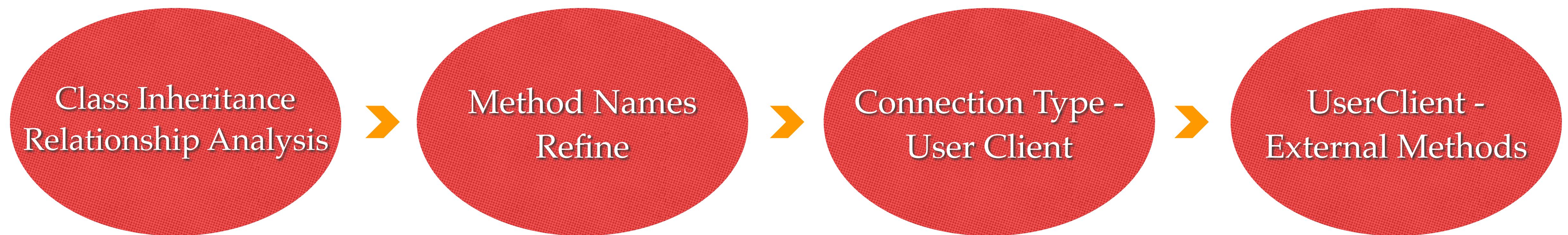
Agenda

- ❖ Static Analysis for Kernel Extensions Attack Interfaces
- ❖ Enhanced PassiveFuzz
- ❖ Vulnerabilities Found
- ❖ Conclusion

Agenda

- ❖ Static Analysis for Kernel Extensions Attack Interfaces
- ❖ Enhanced PassiveFuzz
- ❖ Vulnerabilities Found
- ❖ Conclusion

Kexts Interfaces Analysis Flow



Class Inheritance Relationship

- ❖ rdi/x0: instance of register Meta class
- ❖ rsi/x1: Meta class name
- ❖ rdx/x2: instance of parent Meta class
- ❖ rcx/w3: size of register Meta class instance

```
_GLOBAL__sub_I_IOAccelMemory_cpp proc near
    ; DATA XREF: __mod_init_func:00000000000590E0↓o
    push  rbp
    mov   rbp, rsp
    lea   rdi, __ZN13IOAccelMemory10gMetaClassE ; IOAccelMemory::gMetaClass
    lea   rsi, aIoaccelmemory ; "IOAccelMemory"
    mov   rdx, cs:_ZN80SObject10gMetaClassE_0 ; OSObject::gMetaClass
    mov   ecx, 0A0h ;
    call  __ZN11OSMetaClassC2EPKcPKS_j ; OSMetaClass::OSMetaClass(char const*,OSMetaClass const*,uint)
    lea   rax, off_59550
    mov   cs:_ZN13IOAccelMemory10gMetaClassE, rax ; IOAccelMemory::gMetaClass
    pop   rbp
    retn
_GLOBAL__sub_I_IOAccelMemory_cpp endp
```

Method Name Refine

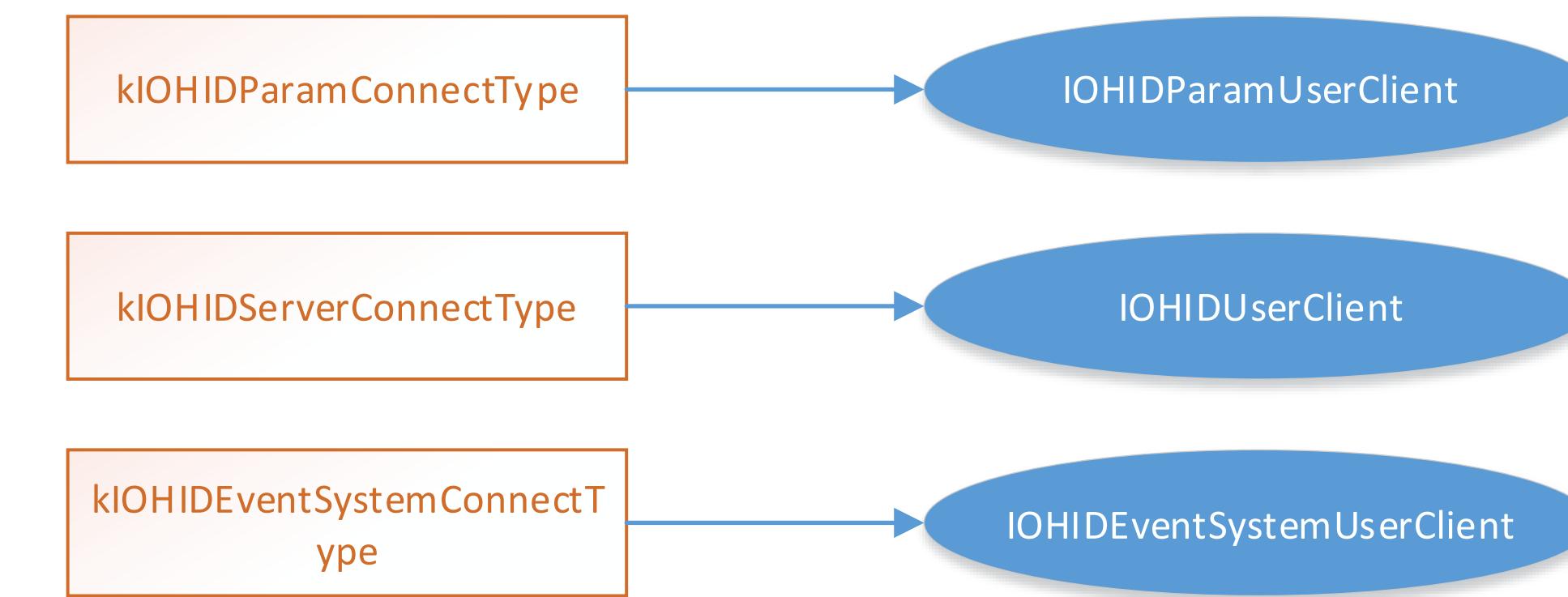
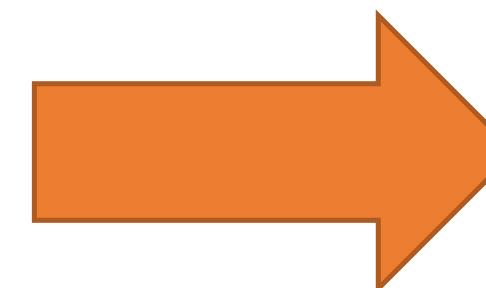
IOMobileFramebuffer -> IOService -> IORegistryEntry -> OSObject

```
ClassName : IOMobileFramebuffer
SuperClass: IOService->IORegistryEntry->OSObject
SuperClass: 0xffffffff00765eb68
ClassSize : 0xdb0
parser.py   sub_0xffffffff0063af65cL
0 : 0xffffffff0063af65cL sub_0xffffffff0063ba7d0L
1 : 0xffffffff0063ba7d0L sub_0xffffffff0063ba7d0L
2 : 0xffffffff00754b618L OSMetaClass::release(int)
3 : 0xffffffff00754b61cL OSMetaClass::getRetainCount()
4 : 0xffffffff00754b624L OSMetaClass::retain()
5 : 0xffffffff00754b628L OSMetaClass::release()
6 : 0xffffffff00754b62cL OSMetaClass::serialize(OSSerialize*)
7 : 0xffffffff00754b64cL OSMetaClass::getMetaClass()
8 : 0xffffffff00754b458L OSMetaClassBase::isEqualTo(OSMetaClassBase const*)
9 : 0xffffffff00754b658L OSMetaClass::taggedRetain(void const*)
10: 0xffffffff00754b65cL OSMetaClass::taggedRelease(void const*)
11: 0xffffffff00754b660L OSMetaClass::taggedRelease(void const*, int)
12: 0xffffffff0063af6d4L sub_0xffffffff0063af6d4L
_____
vtable:0xffffffff006ed14e0I _____
: misc_func.py
IOMobileFramebuffer
0 : 0xffffffff0063af688L sub_0xffffffff0063af688L
1 : 0xffffffff0063af68cL sub_0xffffffff0063af68cL
2 : 0xffffffff00754d4c4L OSObject::release(int)
3 : 0xffffffff00754d4d8L OSObject::getRetainCount()
4 : 0xffffffff00754d4e0L OSObject::retain()
5 : 0xffffffff00754d4f0L OSObject::release()
6 : 0xffffffff00754d500L OSObject::serialize(OSSerialize*)
7 : 0xffffffff0063af690L sub_0xffffffff0063af690L
8 : 0xffffffff00754b458L OSMetaClassBase::isEqualTo(OSMetaClassBase const*)
9 : 0xffffffff00754d5e8L OSObject::taggedRetain(void const*)
10: 0xffffffff00754d680L OSObject::taggedRelease(void const*)
11: 0xffffffff00754d690L OSObject::taggedRelease(void const*, int)
12: 0xffffffff00754d778L OSObject::init()
13: 0xffffffff0063b0118L sub_0xffffffff0063b0118L
_____
misc_func.py
IOService if super_addr in BASE_CLASS:
0 : 0xffffffff00758b1a0L sub_0xffffffff00758b1a0L = BASE_CLASS(super_addr)
1 : IOService::~IOService()
2 : OSObject::release(int)    OSObject::release(int)
3 : OSObject::getRetainCount() OSObject::getRetainCount()
4 : OSObject::retain()       OSObject::retain()
5 : OSObject::release()      OSObject::release()
6 : OSObject::serialize(OSSerialize*) OSObject::serialize(OSSerialize*)
7 : IOService::getMetaClass() name string
8 : OSMetaClassBase::isEqualTo(OSMetaClassBase const*)
9 : OSObject::taggedRetain(void const*) class.instance_list()
10: OSObject::taggedRelease(void const*) class.instance_list[i]
11: OSObject::taggedRelease(void const*, int) OSObject::taggedRelease(void const*, int)
12: OSObject::init()          OSObject::init()
13: IOService::free()         OSObject::free()
_____
IORegistryEntry
sub_0xffffffff007584da8L
IORegistryEntry::~IORegistryEntry()
OSObject::~OSObject()
OSObject::release(int)
OSObject::getRetainCount()
OSObject::retain()
OSObject::release()
OSObject::serialize(OSSerialize*)
IORegistryEntry::getMetaClass()
OSMetaClassBase::isEqualTo(OSMetaClassBase const*)
OSObject::taggedRetain(void const*)
OSObject::taggedRelease(void const*)
OSObject::taggedRelease(void const*, int)
OSObject::init()
IORegistryEntry::free()
_____
OSObject
sub_0xffffffff00754d4c4L
OSObject::~OSObject()
OSObject::release(int)
OSObject::getRetainCount()
OSObject::retain()
OSObject::release()
OSObject::serialize(OSSerialize*)
OSObject::getMetaClass()
OSMetaClassBase::isEqualTo(OSMetaClassBase const*)
OSObject::taggedRetain(void const*)
OSObject::taggedRelease(void const*)
OSObject::taggedRelease(void const*, int)
OSObject::init()
OSObject::free()
```

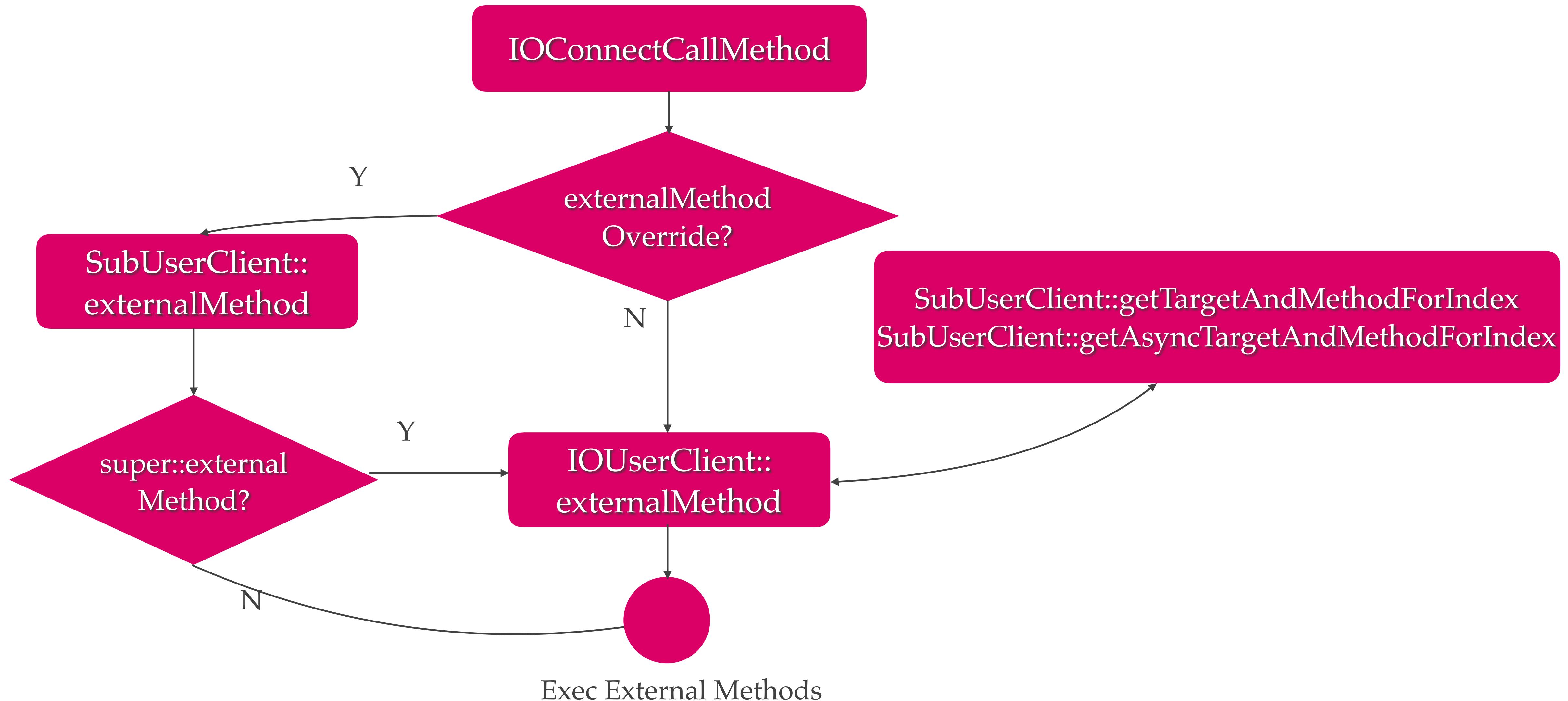
Connection Type - UserClients

- ❖ Locate the newUserClient function address for IOServices
- ❖ Analyze the ASM instructions to enumerate the connection types
- ❖ Analyze the ASM instructions to get the corresponding user client for each connection type

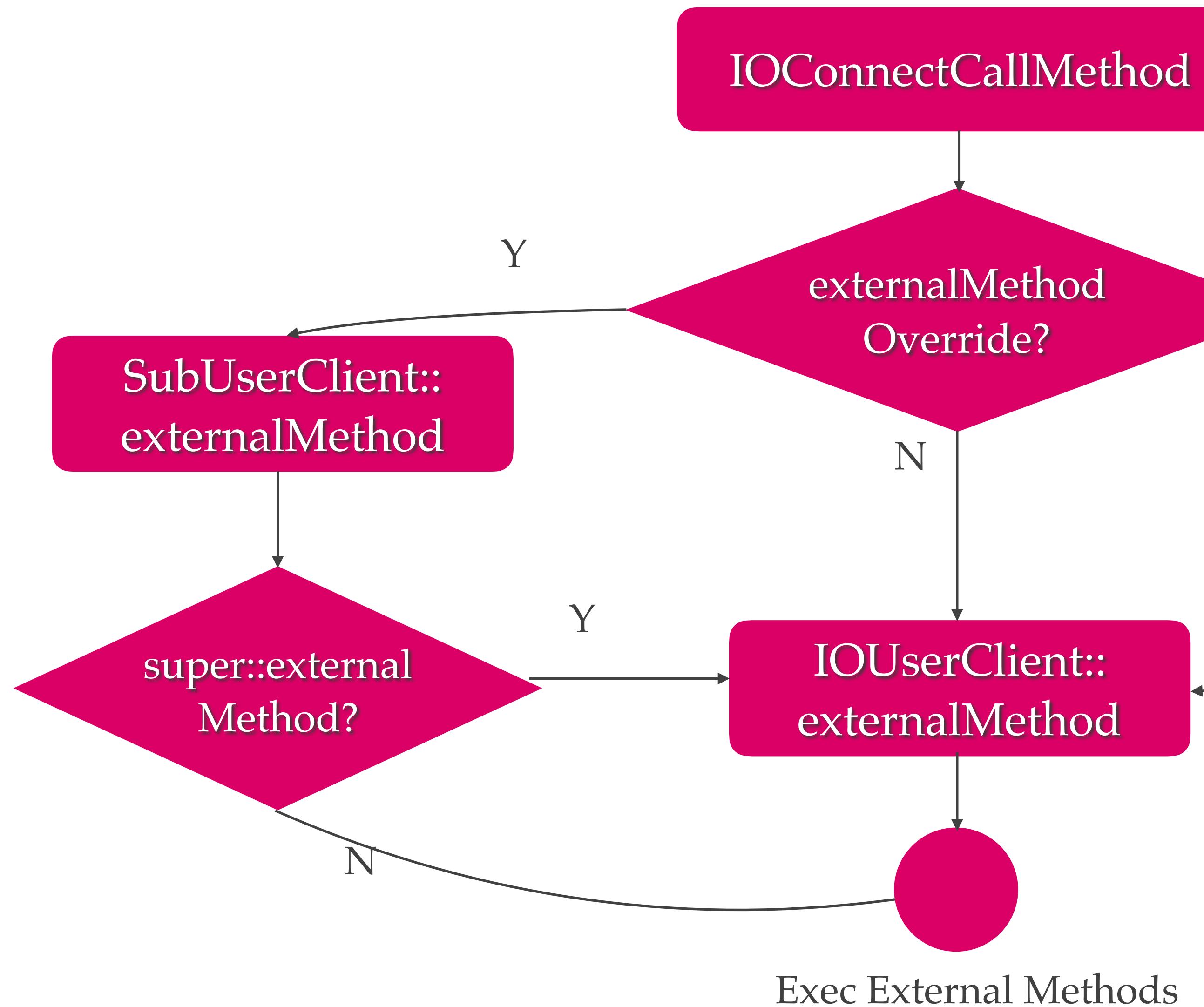
```
do {  
    if (type == kIOHIDParamConnectType) {  
        if (eventsOpen) {  
            newConnect = new IOHIDParamUserClient;  
        } else {  
            err = kIOReturnNotOpen;  
            break;  
        }  
    }  
    else if ( type == kIOHIDServerConnectType) {  
        newConnect = new IOHIDUserClient;  
    }  
    else if ( type == kIOHIDStackShotConnectType ) {  
        newConnect = new IOHIDStackShotUserClient;  
    }  
    else if ( type == kIOHIDEVENTSystemConnectType ) {  
        newConnect = new IOHIDEVENTSystemUserClient;  
    }  
    else {  
        err = kIOReturnUnsupported;  
    }  
}
```



UserClient - External Methods

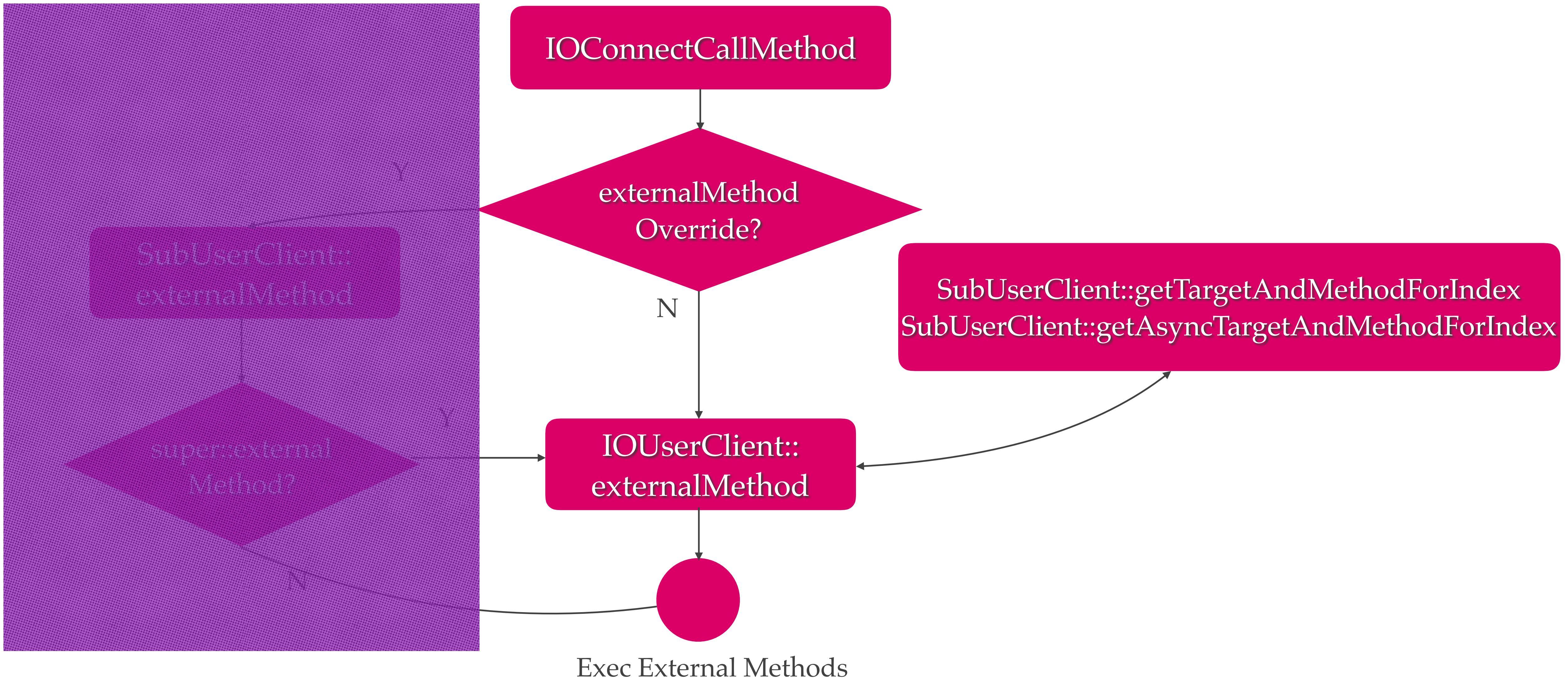


UserClient - External Methods



Exec External Methods

UserClient - External Methods



Two Graceful Implementation(1/3)

```
struct IOExternalMethodDispatch
{
    IOExternalMethodAction function;
    uint32_t          checkScalarInputCount;
    uint32_t          checkStructureInputSize;
    uint32_t          checkScalarOutputCount;
    uint32_t          checkStructureOutputSize;
};
```

```
IOReturn IOHIDEventServiceUserClient::externalMethod(
    uint32_t              selector,
    IOExternalMethodArguments * arguments,
    IOExternalMethodDispatch * dispatch,
    OSObject *             target,
    void *                 reference)
{
    if (selector < (uint32_t) kIOHIDEventServiceUserClientNumCommands)
    {
        dispatch = (IOExternalMethodDispatch *) &sMethods[selector];
        if (!target)
            target = this;
    }
    return super::externalMethod(selector, arguments, dispatch, target, re...
```

Two Graceful Implementation(3/3)

```
//=====
// IOHIDEEventServiceUserClient::sMethods
//=====

const IOExternalMethodDispatch IOHIDEEventServiceUserClient::sMethods[kIOHIDEEventServiceUserClientNumCommands] = {
    { // kIOHIDEEventServiceUserClientOpen
        (IOExternalMethodAction) &IOHIDEEventServiceUserClient::_open,
        1, 0,
        0, 0
    },
    { // kIOHIDEEventServiceUserClientClose
        (IOExternalMethodAction) &IOHIDEEventServiceUserClient::_close,
        1, 0,
        0, 0
    },
    { // kIOHIDEEventServiceUserClientCopyEvent
        (IOExternalMethodAction) &IOHIDEEventServiceUserClient::_copyEvent,
        2, -1,
        0, -1
    },
    { // kIOHIDEEventServiceUserClientSetElementValue
        (IOExternalMethodAction) &IOHIDEEventServiceUserClient::_setElementValue,
        3, 0,
        0, 0
    },
};
```

Two Graceful Implementation(2/3)

```
struct IOExternalMethod
{
    IOService * object;
    IOMethod func;
    IOOptionBits flags;
    IOByteCount count0;
    IOByteCount count1;
};
```

```
IOExternalMethod * IOI2CInterfaceUserClient::getTargetAndMethodForIndex(
    IOService ** targetP, UInt32 index )
{
    static const IOExternalMethod methodTemplate[] = {
        /* 0 */ { NULL, (IOMethod) &IOI2CInterfaceUserClient::extAcquireBus,
                  kIOUCScalarIScalar0, 0, 0 },
        /* 1 */ { NULL, (IOMethod) &IOI2CInterfaceUserClient::extReleaseBus,
                  kIOUCScalarIScalar0, 0, 0 },
        /* 3 */ { NULL, (IOMethod) &IOI2CInterfaceUserClient::extIO,
                  kIOUCStructIStruct0, 0xffffffff, 0xffffffff },
    };
    if (index >= (sizeof(methodTemplate) / sizeof(methodTemplate[0])))
        return (NULL);

    *targetP = this;
    return ((IOExternalMethod *) (methodTemplate + index));
}
```

Parse Method

- ❖ Parse "Symbol Table" section
- ❖ Search Constant Array name, shown as "String Table Index"
- ❖ Start with "_ZN" or "_ZN"
- ❖ Locate the address, shown as "value"

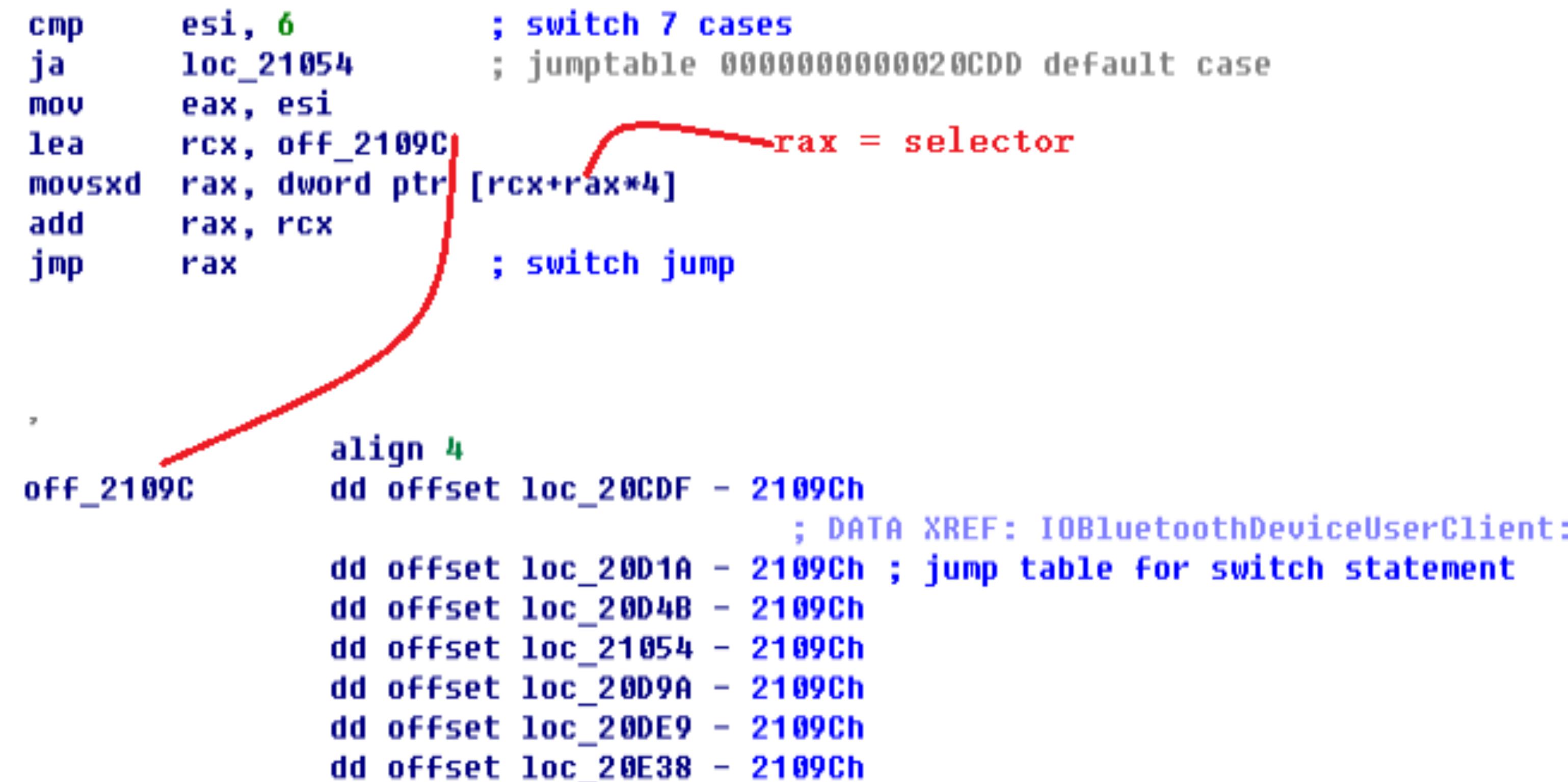
String Table Index	<u>_ZN23I0FramebufferUserClient14externalMethodEjP25I0ExternalMethodArgumentsP24I0ExternalMethodDispatchP80S0bjectPvE14methodTemplate</u>
Type	0E
Section Index	7 (<u>__DATA,__const</u>)
Description	
Value	205360 (\$+41072)
000627A0 0000C05F	String Table Index
000627A4 0F	Type
	0E
	01
000627A5 08	Section Index
000627A6 0000	Description
000627A8 0000000000042F10	Value

The Ugly Implementation

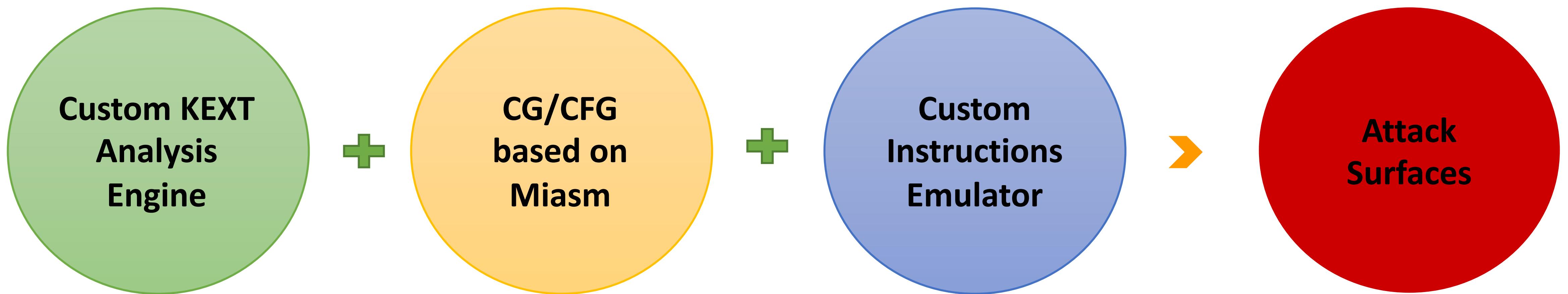
- ❖ Locate the address of override externalMethod Function
- ❖ Analyze the ASM instructions to get selector and external methods

```
    cmp    esi, 6      ; switch 7 cases
    ja     loc_21054   ; jumptable 0000000000020CDD default case
    mov    eax, esi
    lea    rcx, off_2109C
    mousxd rax, dword ptr [rcx+rcx*4]    rax = selector
    add    rax, rcx
    jmp    rax          ; switch jump

    align 4
off_2109C dd offset loc_20CDF - 2109Ch           ; DATA XREF: IOBluetoothDeviceUserClient:
                                                ; jump table for switch statement
    dd offset loc_20D1A - 2109Ch
    dd offset loc_20D4B - 2109Ch
    dd offset loc_21054 - 2109Ch
    dd offset loc_20D9A - 2109Ch
    dd offset loc_20DE9 - 2109Ch
    dd offset loc_20E38 - 2109Ch
```



Analyze the ASM Instructions



Custom KEXTs Analysis Engine

```
▼ c MachOHeader(object)
  m __init__(self, fh, offset, size)
  m get_driver_list(self)
  f __parser_driver_dict(self, bundle)
  m macho_get_vmaddr(self, segname, sectname)
  m macho_get_fileaddr(self, segname, sectname)
  m macho_get_size(self, segname, sectname)
  m macho_get_loadcmds(self)
  m memcpy(self, start_fileaddr, size)
  m get_mem_from_vmaddr(self, anchor_f, anchor_vm, src_vm)
  m get_memStr_from_vmaddr(self, anchor_f, anchor_vm, src_vn)
  m get_memStr_from_f(self, file_off)
  m get_f_from_vm(self, anchor_f, anchor_vm, src_vm)
  m get_vm_from_f(self, anchor_f, anchor_vm, src_f)
  m get_prelinkf_from_vm(self, src_vm)
  m get_prelinkvm_from_f(self, anchor_vm, anchor_f, src_f)
  f MH_MAGIC
  f endian
  f fh
  f kernel_header
  f mach_header
  f offset
  f prelink_offset
  f size
  f sizediff

  ▼ c KernelMachO(object)
    m __init__(self, filename=None, base_addr=0xffffffff00700400)
    m load(self, fh)
    m load_fat(self, fh)
    m load_header(self, fh, offset, size)
    m get_section_addrs(self)
    m get_other_addrs(self)
    m get_driver_list(self)
    m extract_kext(self, bundleID=None, dir=None)
    m __construct_kext(self, bundle, offset, prelink_offset, dir)
    m __dump_kext_data(self, fd, fh_offset, data_size, fd_offset)
    m __parser_driver_dict(self, bundle)
    f base_addr
    f driver_list_notprelink
    f driver_list_prelink
    f fat
    f filename
    f headers

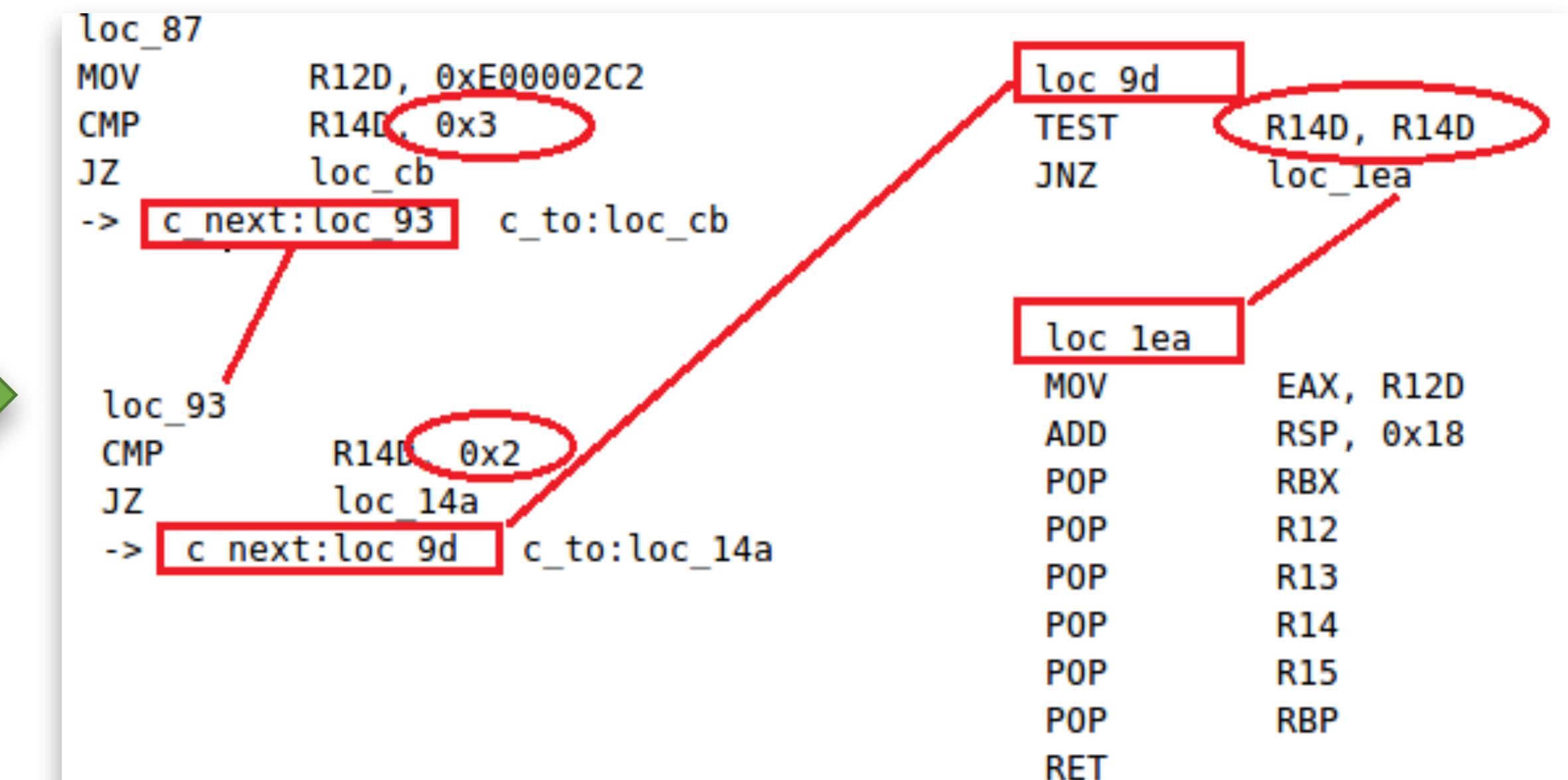
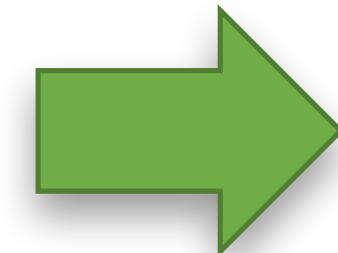
  ▼ c OSMetaClass(object)
    m __init__(self)
    f IOExternalAsyncMethod
    f IOExternalMethod
    f IOExternalMethodDispatch
    f can_ser_open
    f can_ser_open_type
    f class_name
    f class_self_addr
    f class_size
    f class_super_addr
    f class_super_list
    f class_super_name
    f extends_list
    f externalMethod_f
    f externalMethod_vm
    f getAsyncTargetAndMethodForIndex_f
    f getAsyncTargetAndMethodForIndex_vm
    f getTargetAndMethodForIndex_f
    f getTargetAndMethodForIndex_vm
    f getTargetAndTrapForIndex_f
    f getTargetAndTrapForIndex_vm
    f havePublishedResource
    f instance_list
    f is_ioeam
    f is_ioem
    f is_ioemd
    f metaclass_list
    f metaclass_vt_f
    f metaclass_vt_vm
    f newUserClient_f
    f newUserClient_vm
    f object_vt_f
```

Generate CFG using Miasm

AppleHDAEngine::newUserClient(, , type,)

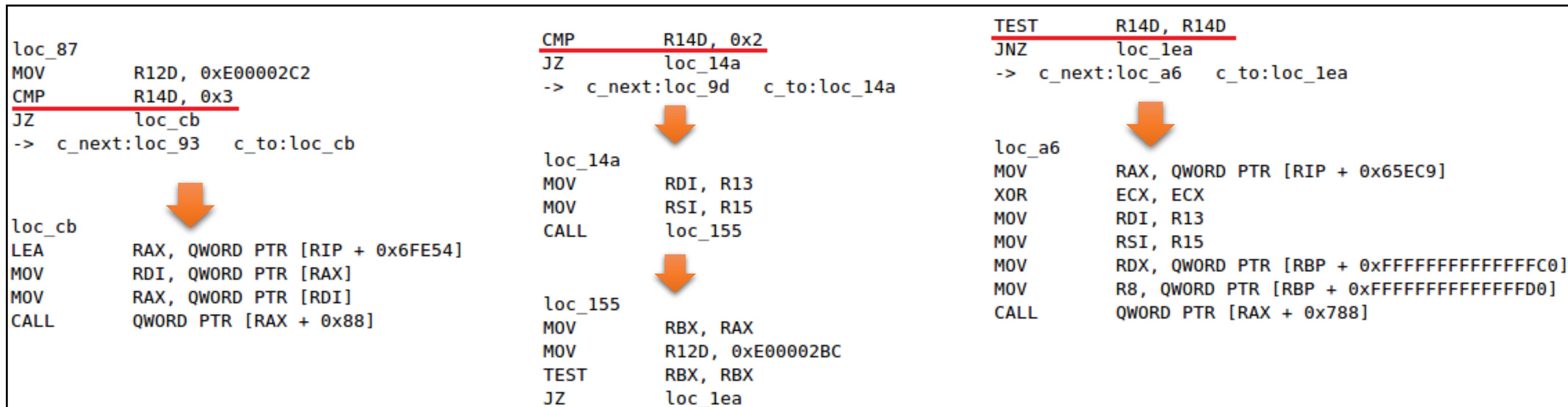
```
loc_2C1F1: ; CODE XREF: AppleHDI
    mov    r12d, 0E00002C2h
    cmp    r14d, 3
    jz     short loc_2C235
    cmp    r14d, 2
    jz     loc_2C2B4
    test   r14d, r14d
    jnz    loc_2C354
    mov    rax, cs:off_920E0
    xor    ecx, ecx
    mov    rdi, r13
    mov    rsi, r15
    mov    rdx, [rbp-40h]
    mov    r8, [rbp-30h]
    call   qword ptr [rax+788h]
    mov    r12d, eax
    jmp    loc_2C354
;

loc_2C235: ; CODE XREF: AppleHDI
    lea    rax, __ZN24AppleHDAEngineUserClient9i
    mov    rdi, [rax]
    mov    rax, [rdi]
    call   qword ptr [rax+88h]
```



Analysis key paths based on CFG

- ❖ Key Paths based on Key registers
 - ❖ RCX register in “newUserClient” function
 - ❖ RSI register in “externalMethod” function
 - ❖ Tracking data flow between registers, as shown below, RCX move to R14D register



Custom Instruction Emulator

❖ ARM Emulator

- ❖ adrp / adr, add, mov / mov

❖ nX86 64 Emulator

- ❖ lea, mov, call, cmp, jz, je...

```
if not cmp(mnemonic, "str"):
    reg_num = insn.op_count(CS_OP_REG)
    if reg_num == 1:
        continue
    f_reg = get_first_reg(insn)
    if f_reg == arm64_const.ARM64_REG_XZR or f_reg == arm64
        f_reg == arm64_const.ARM64_REG_WZR:
            continue
    s_reg = get_second_reg(insn)

    if s_reg:
        s_reg_v = get_actual_value_by_regN(s_reg)
        if not (s_reg_v and s_reg_v == meta_class.class_sel):
            continue
    else:
        continue

    f_reg_v_vm = get_actual_value_by_regN(f_reg)
    if iskext:
        f_reg_v_f = k_header.get_prelinkf_from_vm(f_reg_v_v
    else:
        f_reg_v_f = k_header.get_f_from_vm(each_mif_f, each

parse_const_func(k_header, meta_class, f_reg_v_vm,
                 f_req_v_f, iskext)
```

```
if not cmp(mnemonic, "bl"):
    if insn.op_count(CS_OP_IMM):
        bl_addr_vm = get_single_IMM(insn)
        meta_class = OSMetaClass()
        if bl_addr_vm == OSMetaClass_OSMetaClass_VMAddr:
            #meta_class = OSMetaClass()

def get_single_IMM(insn):
    seg_num = insn.op_count(CS_OP_IMM)
    if seg_num > 1:
        print "Extract: too much imm reg!"
    if seg_num != 1:
        print "Extract: no imm reg found!"
    return to_x(insn.op_find(CS_OP_IMM, 1).value.imm)

def get_mem_op_offset(insn):
    mem_num = insn.op_count(CS_OP_MEM)
    if mem_num >= 1:
        offset = insn.op_find(CS_OP_MEM, 1).mem.disp
    return offset

def get_mem_op_reg(insn):
    mem_num = insn.op_count(CS_OP_MEM)
    if mem_num >= 1:
        offset = insn.op_find(CS_OP_MEM, 1).mem.base
    return offset

def get_first_reg(insn):
    return insn.op_find(CS_OP_REG, 1).value.reg

def get_second_reg(insn):
    return insn.op_find(CS_OP_REG, 2).value.reg

= get_actual_value_by_regN(arm64_const.ARM64_REG_X0)
= get_actual_value_by_regN(arm64_const.ARM64_REG_X1)
r = get_actual_value_by_regN(arm64_const.ARM64_REG_X2)
= get_actual_value_by_regN(arm64_const.ARM64_REG_X3)

ddr:
= k_header.get memStr from vmaddr(each mif f, each mif vm, meta class.class name addr)
class_name,
r = meta_cl
class_name,
meta_class
= "unknown c
ss
6)
m_vm(each_m
r(k_header,
"L") == OSM
= get_actu
= get_actu
r = get_act
t_actual_va
ddr:
= k_header.
class_name,
r = meta_cl
class_name,
meta_class
= "unknown classname"
from capstone import x86_const

class x_reg_manager(object):

    def __init__(self):
        self.x = [1]*234
        for i in range(234):
            self.x[i] = 0

    def get_actual_value_by_regN(self, reg):
        #global x0
        return self.x[reg]

    def set_actual_value_by_regN(self, reg, reg_val):
        self.x[reg] = reg_val
```

Attack Interfaces

AppleHDAEngine::newUserClient

index	CanOpen	TOpenType	ServiceName	extends
4	True	0	AppleHDAEngineOutput	IOAudioEngine::gMetaClass-->AppleHDAEngine-->AppleHDAEngineOutput
86	True	0	AppleHDAEngine	IOAudioEngine::gMetaClass-->AppleHDAEngine
ServiceName	OpenType	UserClient		
AppleHDAEngine	0x3	AppleHDAEngineUserClient::metaClass		
AppleHDAEngine	0x2	DspFuncUserClient::Create(IOAudioEngine*, task*)		

AppleHDAEngineUserClient::externalMethod

selector	cSIC	cSIS	cSOC	cSOS	func_name
0	2	0	0	4095	AppleHDAEngineUserClient::getState
1	2	4095	0	0	AppleHDAEngineUserClient::setState
2	0	0	0	0	AppleHDAEngineUserClient::resetDSPToPropertyList
3	1	0	1	0	AppleHDAEngineUserClient::isPortPresent
4	0	0	6	0	AppleHDAEngineUserClient::getHardwareVolume
5	1	0	0	0	AppleHDAEngineUserClient::setHardwareVolume
6	0	0	16	0	AppleHDAEngineUserClient::getActiveSpatialChannels
7	0	0	3	0	AppleHDAEngineUserClient::getAudioSnoopEnabled
8	3	0	0	0	AppleHDAEngineUserClient::setAudioSnoopEnabled
9	2	0	0	0	AppleHDAEngineUserClient::setSpatialChannelMute

Process finished with exit code 0

Shortage

- ❖ KEXTs are closed source, many method strings are stripped
- ❖ Function call usually use *(object_ptr + offset) type

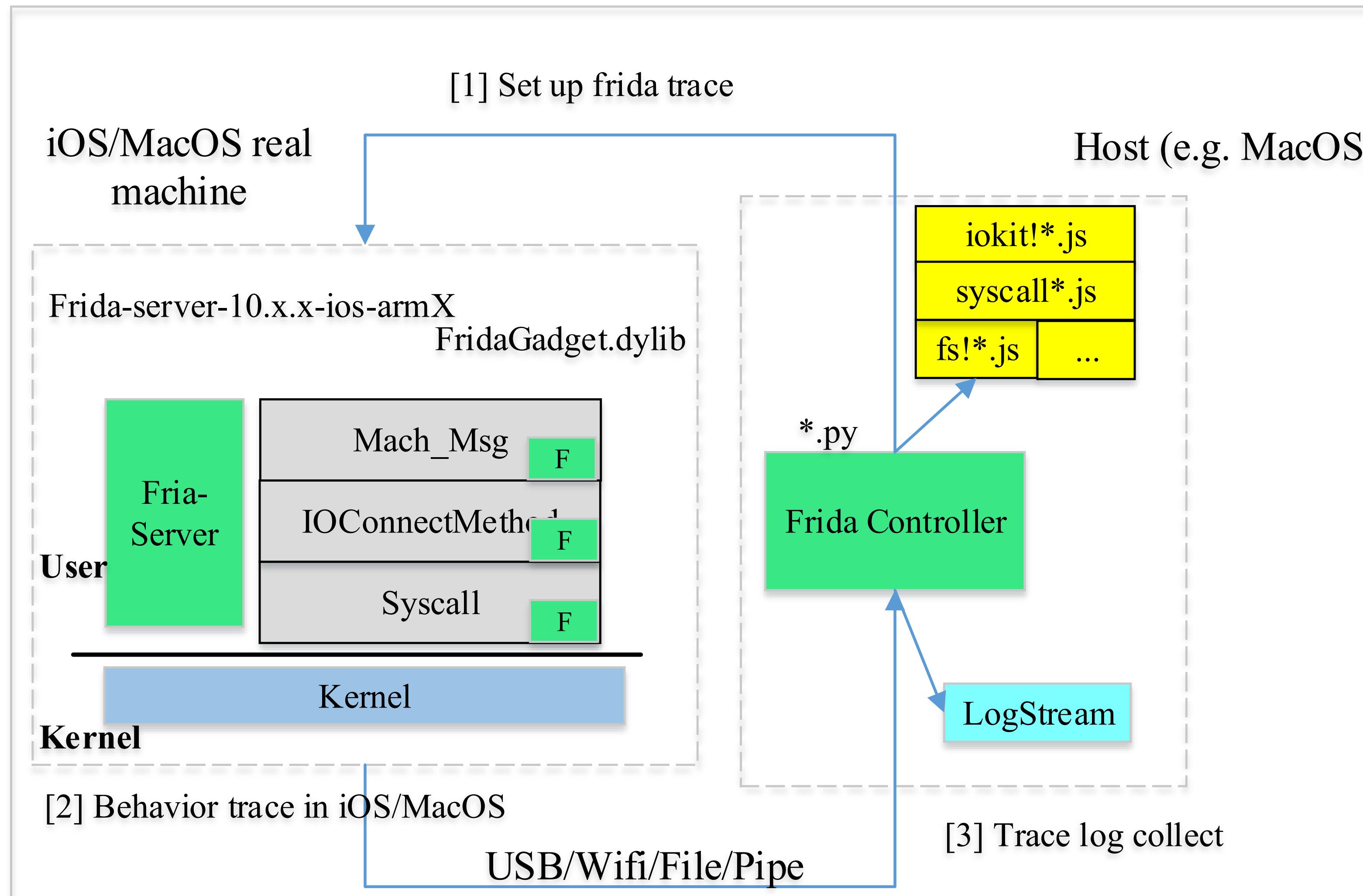
```
v20 = (*(int (__fastcall **)(I0RegistryEntry *, __int64, AMDRadeonX4000_AMDAccelResource *, _QWORD, _QWORD, _QWORD))(*(_QWORD *)this_ptr + 0xB70LL))(  
    this_ptr,  
    v2,  
    accelResource_offset8,  
    0LL,  
    *((_QWORD *)this_ptr + 594),  
    0LL); // AMDRadeonX4000_AMDSIGLContext::bindResource(I0AccelCommandStreamInfo &, I0AccelResource2 *, bool, I0AccelChannel12 *)
```

LLDB Debug is your choose

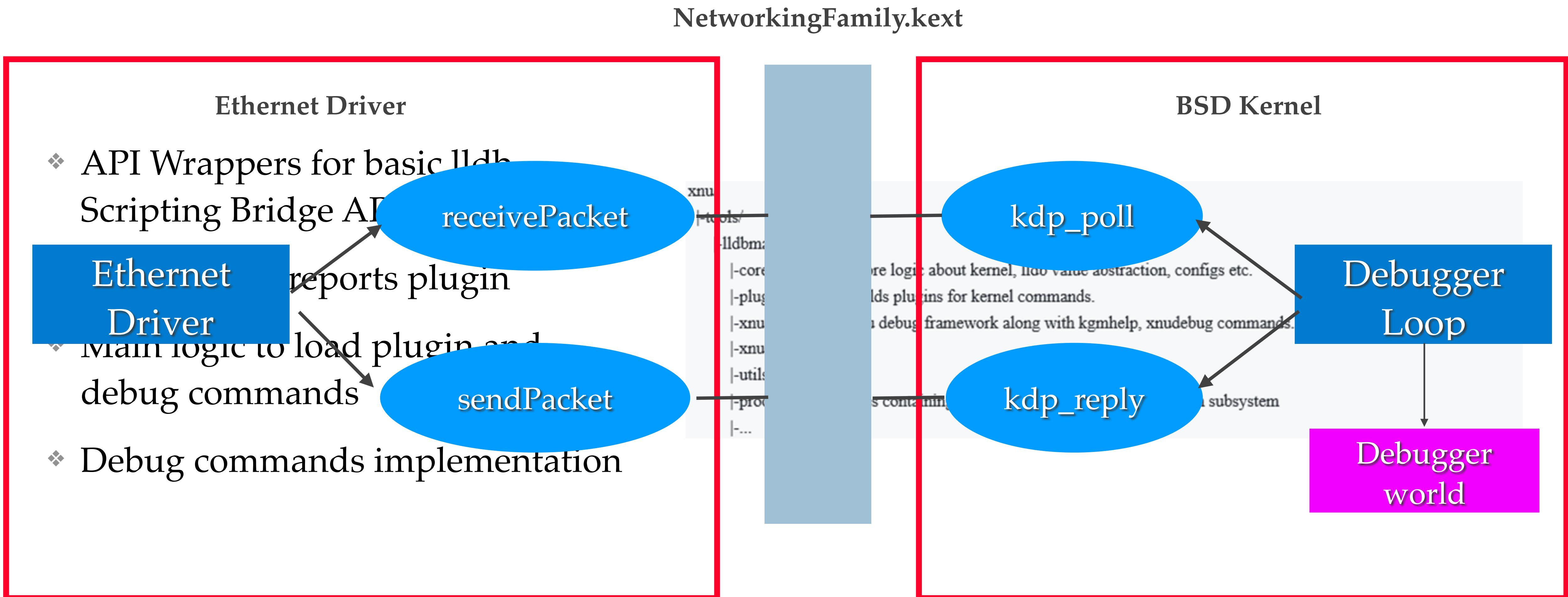
Comparison of dynamic trace

	User Trace	Kernel Trace	Embedded in OS	Any priviledge?	Support script?	Performance	Platform
Frida	Yes	No	No	Root or Repack	Yes	Middle	iOS/Osx
Dtrace	No	Yes	Yes	Root	Yes	High	Osx
lldb	Yes	Yes	Yes	Root	Yes	Low	iOS/Osx
Kernel hook	---	Yes	No	Root	No	Middle	Osx

Frida Hook in User Mode



lldb Kernel Debugging



lldb Kernel Debugging

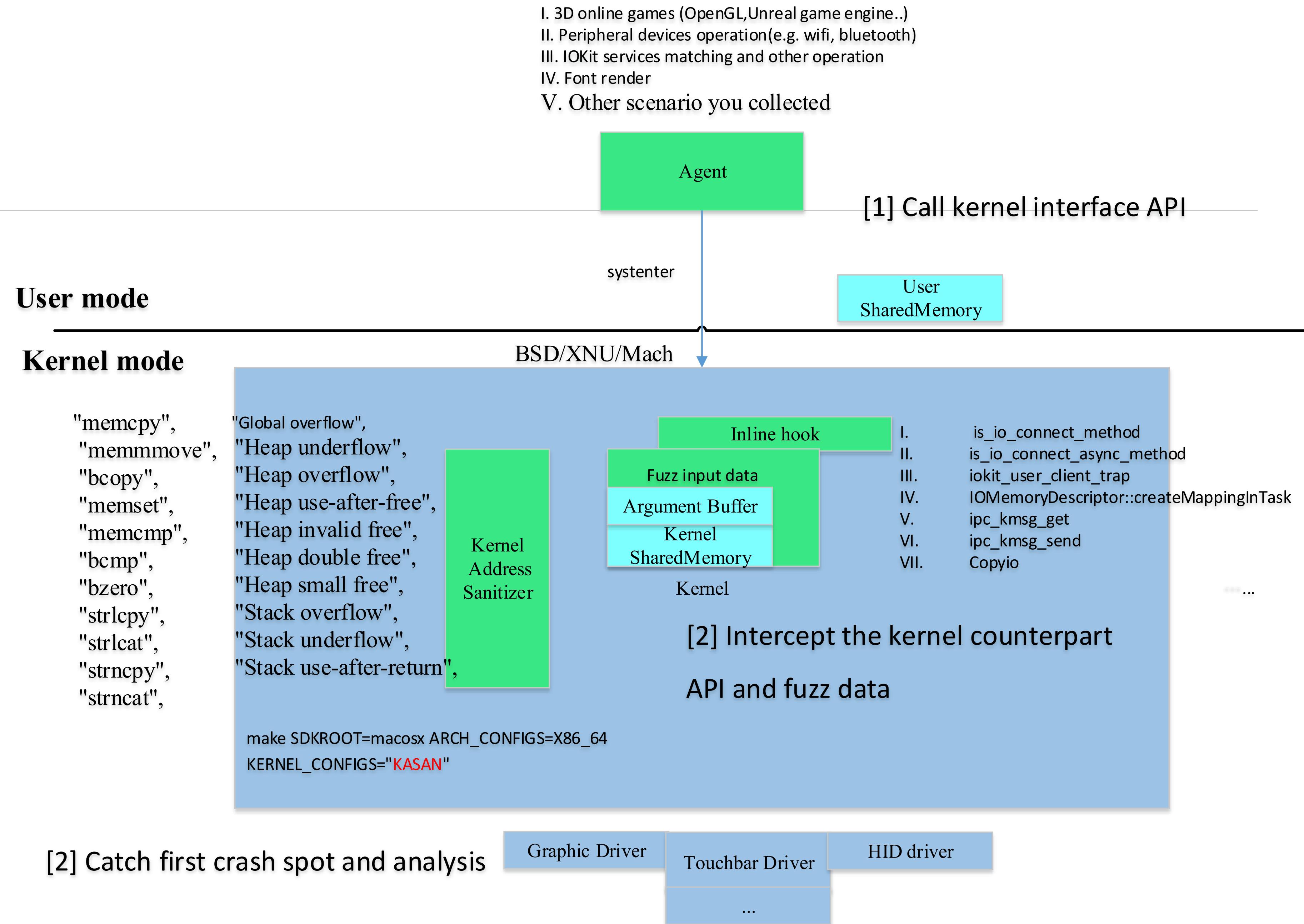
	System mode support	Scriptable	Control Grain	Execution control	Cross platform
DTrace	Kernel	Yes	API	No/View only	Easy
Frida	User	Yes	Instruction	Yes	Easy
Inline hook	Both	No	Instruction	Yes	Middle
LLDB control	Both	Yes	Instruction	Yes	Easy

Agenda

- ❖ Static Analysis for Kernel Extensions Attack Interfaces
- ❖ Enhanced PassiveFuzz
- ❖ Vulnerabilities Found
- ❖ Conclusion

PassiveFuzz

- ❖ Inline HOOK
- ❖ Probe Installation
- ❖ Mutation
- ❖ KASAN
- ❖ Agent
- ❖ Automation



KSAN in XNU kernel

- ❖ make
SDKROOT=macosx
ARCH_CONFIGS=X86_64
KERNEL_CONFIGS="KASAN"

The screenshot shows the IDA Pro interface with the assembly view open. The assembly code is as follows:

```
673     a2 = 8LL;
674     v69 = __asan_report_store_n(v66, 8LL, v14, v67, v63);
675 LABEL_135:
676     if ( (char)((v104 & 7) + 3) < v69 )
677     {
678         LABEL_60:
679         v3 = *(unsigned int *) (v8 + 44);
680         v70 = *(_BYTE *) (a3 - 2305845139517472768LL);
681         if ( v70 )
682             goto LABEL_137;
683         goto LABEL_61;
684     }
685     v70 = __asan_report_load4(v104);
686 LABEL_137:
687     if ( (char)((v102 & 7) + 3) < v70 )
688     {
689         LABEL_61:
690         v3 = ((DWORD)v3 + 3) & 0xFFFFFFFF;
691         a2 = v110;
692         __asan_memcpy(v102, (unsigned int)(8 * *(DWORD *)v102), v8 + v3 + 52, v110);
693         v72 = *(_BYTE *) (v107 - 2305845139517472768LL);
694         if ( v72 )
695             goto LABEL_139;
696         goto LABEL_62;
697     }
698     v72 = __asan_report_load4(v102);
699 LABEL_139:
700     if ( (char)((v102 & 7) + 3) < v72 )
701     {
702         LABEL_62:
703         v73 = v3;
704         a2 = v105 + v3 - 4096;
705         v74 = *(unsigned int *)v102;
```

The assembly code is annotated with several yellow highlights, including:

- Function names: `_Xio_connect_method`, `_Xio_connect_method_scalar0`, `_Xio_connect_method_scalar1`, `_Xio_connect_method_scalar1_structure0`, `_Xio_connect_method_scalar1_structure1`, `_Xio_connect_method_structure0`, `_Xio_connect_method_structure1`, `_Xio_connect_method_var_output`.
- Labels: `LABEL_135`, `LABEL_60`, `LABEL_61`, `LABEL_62`, `LABEL_63`, `LABEL_64`, `LABEL_65`, `LABEL_66`, `LABEL_67`, `LABEL_68`, `LABEL_69`, `LABEL_70`, `LABEL_71`, `LABEL_72`, `LABEL_73`, `LABEL_74`, `LABEL_75`, `LABEL_76`, `LABEL_77`, `LABEL_78`, `LABEL_79`, `LABEL_80`, `LABEL_81`, `LABEL_82`, `LABEL_83`, `LABEL_84`, `LABEL_85`, `LABEL_86`, `LABEL_87`, `LABEL_88`, `LABEL_89`, `LABEL_90`, `LABEL_91`, `LABEL_92`, `LABEL_93`, `LABEL_94`, `LABEL_95`, `LABEL_96`, `LABEL_97`, `LABEL_98`, `LABEL_99`, `LABEL_100`, `LABEL_101`, `LABEL_102`, `LABEL_103`, `LABEL_104`, `LABEL_105`, `LABEL_106`, `LABEL_107`, `LABEL_108`, `LABEL_109`, `LABEL_110`, `LABEL_111`, `LABEL_112`, `LABEL_113`, `LABEL_114`, `LABEL_115`, `LABEL_116`, `LABEL_117`, `LABEL_118`, `LABEL_119`, `LABEL_120`, `LABEL_121`, `LABEL_122`, `LABEL_123`, `LABEL_124`, `LABEL_125`, `LABEL_126`, `LABEL_127`, `LABEL_128`, `LABEL_129`, `LABEL_130`, `LABEL_131`, `LABEL_132`, `LABEL_133`, `LABEL_134`, `LABEL_135`, `LABEL_136`, `LABEL_137`, `LABEL_138`, `LABEL_139`, `LABEL_140`, `LABEL_141`, `LABEL_142`, `LABEL_143`, `LABEL_144`, `LABEL_145`, `LABEL_146`, `LABEL_147`, `LABEL_148`, `LABEL_149`, `LABEL_150`, `LABEL_151`, `LABEL_152`, `LABEL_153`, `LABEL_154`, `LABEL_155`, `LABEL_156`, `LABEL_157`, `LABEL_158`, `LABEL_159`, `LABEL_160`, `LABEL_161`, `LABEL_162`, `LABEL_163`, `LABEL_164`, `LABEL_165`, `LABEL_166`, `LABEL_167`, `LABEL_168`, `LABEL_169`, `LABEL_170`, `LABEL_171`, `LABEL_172`, `LABEL_173`, `LABEL_174`, `LABEL_175`, `LABEL_176`, `LABEL_177`, `LABEL_178`, `LABEL_179`, `LABEL_180`, `LABEL_181`, `LABEL_182`, `LABEL_183`, `LABEL_184`, `LABEL_185`, `LABEL_186`, `LABEL_187`, `LABEL_188`, `LABEL_189`, `LABEL_190`, `LABEL_191`, `LABEL_192`, `LABEL_193`, `LABEL_194`, `LABEL_195`, `LABEL_196`, `LABEL_197`, `LABEL_198`, `LABEL_199`, `LABEL_200`, `LABEL_201`, `LABEL_202`, `LABEL_203`, `LABEL_204`, `LABEL_205`, `LABEL_206`, `LABEL_207`, `LABEL_208`, `LABEL_209`, `LABEL_210`, `LABEL_211`, `LABEL_212`, `LABEL_213`, `LABEL_214`, `LABEL_215`, `LABEL_216`, `LABEL_217`, `LABEL_218`, `LABEL_219`, `LABEL_220`, `LABEL_221`, `LABEL_222`, `LABEL_223`, `LABEL_224`, `LABEL_225`, `LABEL_226`, `LABEL_227`, `LABEL_228`, `LABEL_229`, `LABEL_230`, `LABEL_231`, `LABEL_232`, `LABEL_233`, `LABEL_234`, `LABEL_235`, `LABEL_236`, `LABEL_237`, `LABEL_238`, `LABEL_239`, `LABEL_240`, `LABEL_241`, `LABEL_242`, `LABEL_243`, `LABEL_244`, `LABEL_245`, `LABEL_246`, `LABEL_247`, `LABEL_248`, `LABEL_249`, `LABEL_250`, `LABEL_251`, `LABEL_252`, `LABEL_253`, `LABEL_254`, `LABEL_255`, `LABEL_256`, `LABEL_257`, `LABEL_258`, `LABEL_259`, `LABEL_260`, `LABEL_261`, `LABEL_262`, `LABEL_263`, `LABEL_264`, `LABEL_265`, `LABEL_266`, `LABEL_267`, `LABEL_268`, `LABEL_269`, `LABEL_270`, `LABEL_271`, `LABEL_272`, `LABEL_273`, `LABEL_274`, `LABEL_275`, `LABEL_276`, `LABEL_277`, `LABEL_278`, `LABEL_279`, `LABEL_280`, `LABEL_281`, `LABEL_282`, `LABEL_283`, `LABEL_284`, `LABEL_285`, `LABEL_286`, `LABEL_287`, `LABEL_288`, `LABEL_289`, `LABEL_290`, `LABEL_291`, `LABEL_292`, `LABEL_293`, `LABEL_294`, `LABEL_295`, `LABEL_296`, `LABEL_297`, `LABEL_298`, `LABEL_299`, `LABEL_300`, `LABEL_301`, `LABEL_302`, `LABEL_303`, `LABEL_304`, `LABEL_305`, `LABEL_306`, `LABEL_307`, `LABEL_308`, `LABEL_309`, `LABEL_310`, `LABEL_311`, `LABEL_312`, `LABEL_313`, `LABEL_314`, `LABEL_315`, `LABEL_316`, `LABEL_317`, `LABEL_318`, `LABEL_319`, `LABEL_320`, `LABEL_321`, `LABEL_322`, `LABEL_323`, `LABEL_324`, `LABEL_325`, `LABEL_326`, `LABEL_327`, `LABEL_328`, `LABEL_329`, `LABEL_330`, `LABEL_331`, `LABEL_332`, `LABEL_333`, `LABEL_334`, `LABEL_335`, `LABEL_336`, `LABEL_337`, `LABEL_338`, `LABEL_339`, `LABEL_340`, `LABEL_341`, `LABEL_342`, `LABEL_343`, `LABEL_344`, `LABEL_345`, `LABEL_346`, `LABEL_347`, `LABEL_348`, `LABEL_349`, `LABEL_350`, `LABEL_351`, `LABEL_352`, `LABEL_353`, `LABEL_354`, `LABEL_355`, `LABEL_356`, `LABEL_357`, `LABEL_358`, `LABEL_359`, `LABEL_360`, `LABEL_361`, `LABEL_362`, `LABEL_363`, `LABEL_364`, `LABEL_365`, `LABEL_366`, `LABEL_367`, `LABEL_368`, `LABEL_369`, `LABEL_370`, `LABEL_371`, `LABEL_372`, `LABEL_373`, `LABEL_374`, `LABEL_375`, `LABEL_376`, `LABEL_377`, `LABEL_378`, `LABEL_379`, `LABEL_380`, `LABEL_381`, `LABEL_382`, `LABEL_383`, `LABEL_384`, `LABEL_385`, `LABEL_386`, `LABEL_387`, `LABEL_388`, `LABEL_389`, `LABEL_390`, `LABEL_391`, `LABEL_392`, `LABEL_393`, `LABEL_394`, `LABEL_395`, `LABEL_396`, `LABEL_397`, `LABEL_398`, `LABEL_399`, `LABEL_400`, `LABEL_401`, `LABEL_402`, `LABEL_403`, `LABEL_404`, `LABEL_405`, `LABEL_406`, `LABEL_407`, `LABEL_408`, `LABEL_409`, `LABEL_410`, `LABEL_411`, `LABEL_412`, `LABEL_413`, `LABEL_414`, `LABEL_415`, `LABEL_416`, `LABEL_417`, `LABEL_418`, `LABEL_419`, `LABEL_420`, `LABEL_421`, `LABEL_422`, `LABEL_423`, `LABEL_424`, `LABEL_425`, `LABEL_426`, `LABEL_427`, `LABEL_428`, `LABEL_429`, `LABEL_430`, `LABEL_431`, `LABEL_432`, `LABEL_433`, `LABEL_434`, `LABEL_435`, `LABEL_436`, `LABEL_437`, `LABEL_438`, `LABEL_439`, `LABEL_440`, `LABEL_441`, `LABEL_442`, `LABEL_443`, `LABEL_444`, `LABEL_445`, `LABEL_446`, `LABEL_447`, `LABEL_448`, `LABEL_449`, `LABEL_450`, `LABEL_451`, `LABEL_452`, `LABEL_453`, `LABEL_454`, `LABEL_455`, `LABEL_456`, `LABEL_457`, `LABEL_458`, `LABEL_459`, `LABEL_460`, `LABEL_461`, `LABEL_462`, `LABEL_463`, `LABEL_464`, `LABEL_465`, `LABEL_466`, `LABEL_467`, `LABEL_468`, `LABEL_469`, `LABEL_470`, `LABEL_471`, `LABEL_472`, `LABEL_473`, `LABEL_474`, `LABEL_475`, `LABEL_476`, `LABEL_477`, `LABEL_478`, `LABEL_479`, `LABEL_480`, `LABEL_481`, `LABEL_482`, `LABEL_483`, `LABEL_484`, `LABEL_485`, `LABEL_486`, `LABEL_487`, `LABEL_488`, `LABEL_489`, `LABEL_490`, `LABEL_491`, `LABEL_492`, `LABEL_493`, `LABEL_494`, `LABEL_495`, `LABEL_496`, `LABEL_497`, `LABEL_498`, `LABEL_499`, `LABEL_500`, `LABEL_501`, `LABEL_502`, `LABEL_503`, `LABEL_504`, `LABEL_505`, `LABEL_506`, `LABEL_507`, `LABEL_508`, `LABEL_509`, `LABEL_510`, `LABEL_511`, `LABEL_512`, `LABEL_513`, `LABEL_514`, `LABEL_515`, `LABEL_516`, `LABEL_517`, `LABEL_518`, `LABEL_519`, `LABEL_520`, `LABEL_521`, `LABEL_522`, `LABEL_523`, `LABEL_524`, `LABEL_525`, `LABEL_526`, `LABEL_527`, `LABEL_528`, `LABEL_529`, `LABEL_530`, `LABEL_531`, `LABEL_532`, `LABEL_533`, `LABEL_534`, `LABEL_535`, `LABEL_536`, `LABEL_537`, `LABEL_538`, `LABEL_539`, `LABEL_540`, `LABEL_541`, `LABEL_542`, `LABEL_543`, `LABEL_544`, `LABEL_545`, `LABEL_546`, `LABEL_547`, `LABEL_548`, `LABEL_549`, `LABEL_550`, `LABEL_551`, `LABEL_552`, `LABEL_553`, `LABEL_554`, `LABEL_555`, `LABEL_556`, `LABEL_557`, `LABEL_558`, `LABEL_559`, `LABEL_560`, `LABEL_561`, `LABEL_562`, `LABEL_563`, `LABEL_564`, `LABEL_565`, `LABEL_566`, `LABEL_567`, `LABEL_568`, `LABEL_569`, `LABEL_570`, `LABEL_571`, `LABEL_572`, `LABEL_573`, `LABEL_574`, `LABEL_575`, `LABEL_576`, `LABEL_577`, `LABEL_578`, `LABEL_579`, `LABEL_580`, `LABEL_581`, `LABEL_582`, `LABEL_583`, `LABEL_584`, `LABEL_585`, `LABEL_586`, `LABEL_587`, `LABEL_588`, `LABEL_589`, `LABEL_590`, `LABEL_591`, `LABEL_592`, `LABEL_593`, `LABEL_594`, `LABEL_595`, `LABEL_596`, `LABEL_597`, `LABEL_598`, `LABEL_599`, `LABEL_600`, `LABEL_601`, `LABEL_602`, `LABEL_603`, `LABEL_604`, `LABEL_605`, `LABEL_606`, `LABEL_607`, `LABEL_608`, `LABEL_609`, `LABEL_610`, `LABEL_611`, `LABEL_612`, `LABEL_613`, `LABEL_614`, `LABEL_615`, `LABEL_616`, `LABEL_617`, `LABEL_618`, `LABEL_619`, `LABEL_620`, `LABEL_621`, `LABEL_622`, `LABEL_623`, `LABEL_624`, `LABEL_625`, `LABEL_626`, `LABEL_627`, `LABEL_628`, `LABEL_629`, `LABEL_630`, `LABEL_631`, `LABEL_632`, `LABEL_633`, `LABEL_634`, `LABEL_635`, `LABEL_636`, `LABEL_637`, `LABEL_638`, `LABEL_639`, `LABEL_640`, `LABEL_641`, `LABEL_642`, `LABEL_643`, `LABEL_644`, `LABEL_645`, `LABEL_646`, `LABEL_647`, `LABEL_648`, `LABEL_649`, `LABEL_650`, `LABEL_651`, `LABEL_652`, `LABEL_653`, `LABEL_654`, `LABEL_655`, `LABEL_656`, `LABEL_657`, `LABEL_658`, `LABEL_659`, `LABEL_660`, `LABEL_661`, `LABEL_662`, `LABEL_663`, `LABEL_664`, `LABEL_665`, `LABEL_666`, `LABEL_667`, `LABEL_668`, `LABEL_669`, `LABEL_670`, `LABEL_671`, `LABEL_672`, `LABEL_673`, `LABEL_674`, `LABEL_675`, `LABEL_676`, `LABEL_677`, `LABEL_678`, `LABEL_679`, `LABEL_680`, `LABEL_681`, `LABEL_682`, `LABEL_683`, `LABEL_684`, `LABEL_685`, `LABEL_686`, `LABEL_687`, `LABEL_688`, `LABEL_689`, `LABEL_690`, `LABEL_691`, <

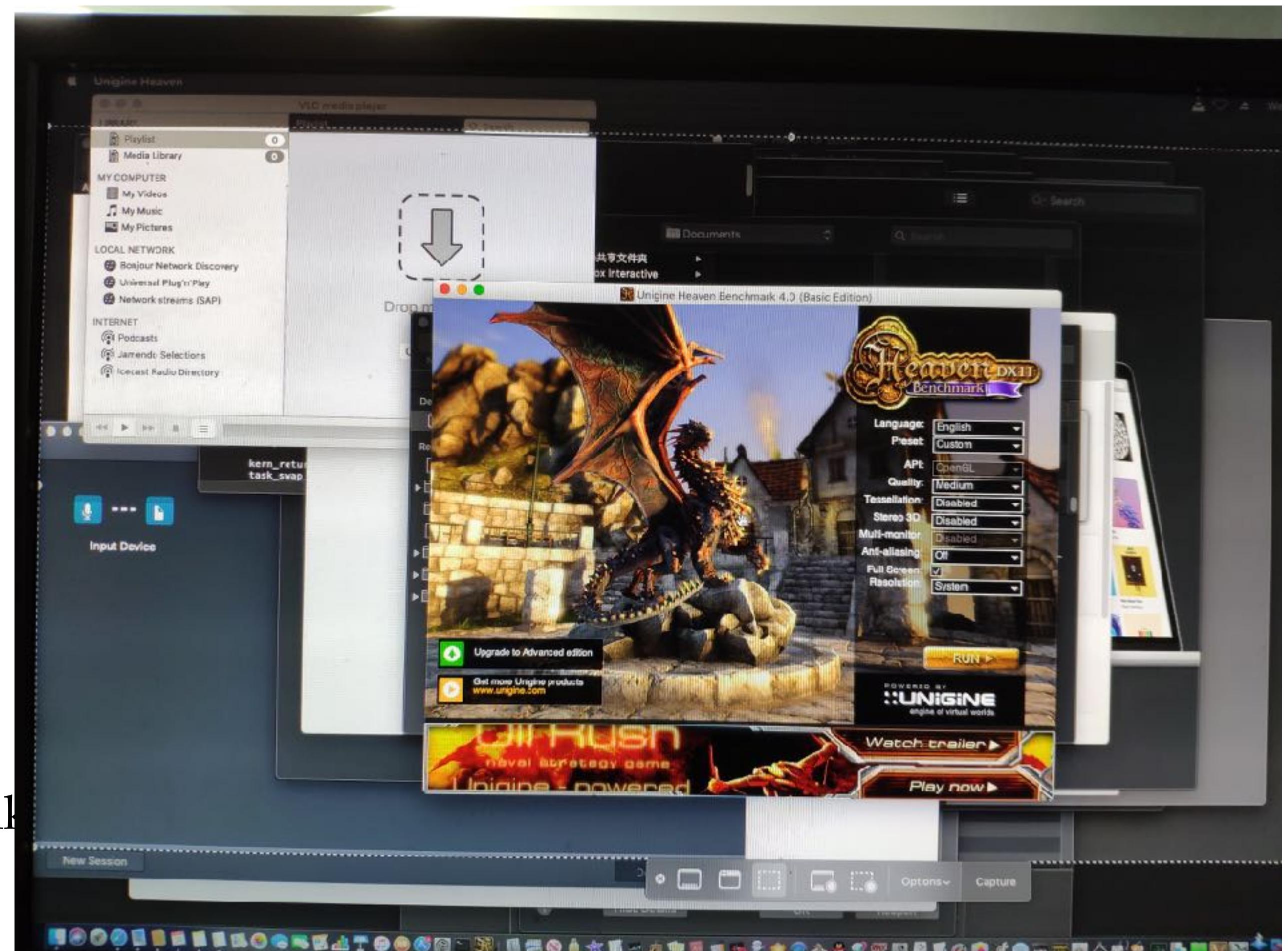
Crash Monitor

- ❖ Embedded llDb toolset is power
- ❖ Fruitful python plugin

```
1  # 
2  def diagnoseOnce(targetIP):
3      settingSymbolResult = settingSymbols()
4      remoteResult = kdpRemote(targetIP)
5      if not checkKdpRemoteConnected():
6          return
7
8      btResult = getCallStack()
9      conAddr = getConnectionAddrFromBt(btResult)
10     showObjResult = showObject(conAddr)
11     objName = getObjNameFromShowObj(showObjResult)
12     selector = getSelectorFromBt(btResult)
13     disResult = getDisAssemble()
14     regsResult = getRegs()
15
16     kdpRemote(targetIP)
17     coreResult = sendCore(dumpServerIP, coreName)
18     doDetach()
19     monitorFileUntilDone(coreFilePath)
20
21     print("diagnoseOnce exit")
22
23     def startRFCLoopInterface(debugger, targetIP, result, internal_dict):
24         counter = 0
25
26         while True:
27             try:
28                 diagnoseOnce()
29             except Exception as e:
30                 print (colored("[ERROR] " + str(e), "red"))
31                 traceback.print_exc()
32                 counter = counter + 1
33                 #break
34
35     # And the initialization code to add your commands
36     def __lldb_init_module(debugger, internal_dict):
37         debugger.HandleCommand('command script add rfc -f remoteFuzzController.startRFCLoopInterface')
```

Data Generating on Target Machine

- ❖ Purpose
 - ❖ Touch more deep code coverage
- ❖ Methodology
 - ❖ Generating valid data and code execution (with context)
- ❖ Implementation
 - ❖ Apple Script based app test
 - ❖ Enrich kinds of corpuses around attack interface
 - ❖ Browser, 3D game engine, benchmark ...as you think



Data Generating Tricks

- ❖ Embedded Apple Script is powerful
- ❖ Automatic app test
- ❖ Timely reboot from kernel
 - ❖ In case of kernel hang but not crash

```
1 void doReboot()
2 {
3     fnPEHaltRestart afnPEHaltRestart = NULL;
4     fnhalt_all_cpus afnhalt_all_cpus = NULL;
5     afnPEHaltRestart = (fnPEHaltRestart) solve_kernel_symbol(&g_kernel_info, "PEHaltRestart");
6     afnPEHaltRestart(kPERestartCPU);
7     afnhalt_all_cpus = (fnhalt_all_cpus)solve_kernel_symbol(&g_kernel_info, "halt_all_cpus");
8     afnhalt_all_cpus(TRUE);
9 }
10 void watchdogTimelyRebootThread(__unused void *arg, __unused wait_result_t wr)
11 {
12     unsigned int nCountSeconds = (unsigned int)arg;
13     struct timespec ts = { nCountSeconds, 0 };
14     int error = 0;
15     lck_mtx_lock(watch_dogt_timely_reboot_mutex);
16     while (1) {
17         aioSleep aioSleep = (fnIOSleep)solve_kernel_symbol(&g_kernel_info, API_SYMBOL_IO_SLEEP);
18         aioSleep(nCountSeconds+1000);
19         printf("[DEBUG] doReboot: end...\n");
20         doReboot();
21         //doCheck();
22     }
23 }
24 kern_return_t startWatdogForTimelyReboot(unsigned int nCountSeconds)
25 {
26     printf("[DEBUG] startWatdogForTimelyReboot (%d): begin...\n", nCountSeconds);
27     kern_return_t kr = KERN_SUCCESS;
28     watch_dogt_timely_reboot_grp = lck_grp_alloc_init("startWatdogForTimelyReboot", LCK_GRP_ATTR_NULL);
29     watch_dogt_timely_reboot_mutex = lck_mtx_alloc_init(watch_dogt_timely_reboot_grp, LCK_ATTR_NULL);
30     kr = kernel_thread_start(watchdogTimelyRebootThread, (void *)nCountSeconds, &tWatchdogThreadHandle);
31     printf("[DEBUG] startWatdogForTimelyReboot (%d): end...\n", nCountSeconds);
32     return kr;
33 }
```

```
170 StartupFuzzingSource()
171 {
172     echo "[StartupFuzzingSource]: begin..."
173     startTimelyRebootDriver $sleepSecondsOfSession
174
175     for counter in $(seq 1 999999)
176     do
177         #Test vm
178         openAnyFolder "/Users/user/Desktop/VM/CDOS_VB/CDOS*/*.vbox"
179         openAnyFolder "/Users/user/Desktop/VM/CDOS_VM/CDOS*/*.vmx"
180         openAnyFolder "/Users/user/Desktop/VM/osx*/*.vmx"
181
182         #External device test zone
183         facetimeTest
184         VLCTest
185
186         #Safari browser test zone
187         safariTest
188
189         #Game engine test zone
190         heavenBenchMarkTest
191         valleyBenchMarkTest
192         heavenBenchMarkTest
193
194         #Misc test zone
195         openAnyFolder "/Applications/*"
196         openAnyFolder "/Users/user/Downloads/*"
197         openAnyFolder "/Applications/Utilities/*"
198         sleep $sleepSecondsInFuzz
199
200         if [ "$counter" -ge "$counterToReboot" ]
201         then
202             echo "now reboot"
203             rebootNow
204         fi
205     done
206     echo "[StartupFuzzingSource]: end..."
207 }
208
209 commonTestApp()
210 {
211     appName=$1
212     #appName="FaceTime"
213     sudo killall -9 $appName
214     osascript -e 'on run {appNameArg}' -e "quit app appNameArg" -e 'end run' $appName
215
216     osascript -e 'on run {appNameArg}' -e "tell application appNameArg to activate" -e 'end run' $appName
217     echo "[commonTestApp]:\tfuzz FaceTime done..."
218 }
```

Agenda

- ❖ Static Analysis for Kernel Extensions Attack Interfaces
- ❖ Enhanced PassiveFuzz
- ❖ Vulnerabilities Found
- ❖ Conclusion

CVE-2018-4462

- ❖ Integer overflow vulnerability in AMDFramebuffer driver

```
[lldb) bt
* thread #1, stop reason = signal SIGSTOP
  * frame #0: 0xffffffff7f8d91e324 AMDFramebuffer`AMDFramebuffer::getPixelInformationFromTiming(AtiDetailedTimingInformation const&, IOPixelInformation*, int, int) + 388
    frame #1: 0xffffffff7f8d91e180 AMDFramebuffer`AMDFramebuffer::getPixelInformation(int, int, IOPixelInformation*) + 112
    frame #2: 0xffffffff7f8d91e0a5 AMDFramebuffer`AMDFramebuffer::getPixelInformation(int, int, int, IOPixelInformation*) + 101
    frame #3: 0xffffffff7f8b42223d IOGraphicsFamily`IOFramebuffer::extGetPixelInformation(target=0xffffffff869e59f000, reference=<unavailable>, args=<unavailable>) at IOFramebu
    frame #4: 0xfffffff800aa4c478 kernel.development`IOUserClient::externalMethod(this=<unavailable>, selector=<unavailable>, args=0xffffffa756c8b988, dispatch=0xffffffff7f8
0000) at IOUserClient.cpp:5335 [opt]
    frame #5: 0xffffffff7f8b437d0b IOGraphicsFamily`IOFramebufferUserClient::externalMethod(this=0xfffffff80b8810800, selector=1, args=0xffffffa756c8b988, dispatch=<unavaila
amebufferUserClient.cpp:380 [opt]
    frame #6: 0xfffffff800aa553cf kernel.development`::is_io_connect_method(connection=0xfffffff80b8810800, selector=1, scalar_input=<unavailable>, scalar_inputCnt=<unavai
put=0, ool_input_size=0, inband_output="", inband_outputCnt=0xfffffff80ac2e2e0c, scalar_output=0xffffffa756c8bcb0, scalar_outputCnt=0xffffffa756c8bcac, ool_output=0, ool_o
]
    frame #7: 0xffffffff7f8e6c854b pasive_kernel_fuzz`trampline_is_io_connect_method(connection=0xfffffff80b8810800, selector=1, scalar_input=0xfffffff80b8b81e10, scalar_inpu
_input_size=0, inband_output="", inband_outputCnt=0xfffffff80ac2e2e0c, scalar_output=0xffffffa756c8bcb0, scalar_outputCnt=0xffffffa756c8bcac, ool_output=0, ool_output_size
    frame #8: 0xfffffff800a3f2bd4 kernel.development`_Xio_connect_method(InHeadP=<unavailable>, OutHeadP=0xfffffff80ac2e2de0) at device_server.c:8379 [opt]
    frame #9: 0xfffffff800a2c450d kernel.development`ipc_kobject_server(request=0xfffffff80b8b81d70, option=<unavailable>) at ipc_kobject.c:359 [opt]
    frame #10: 0xfffffff800a29124a kernel.development`ipc_kmsg_send(kmsg=0xfffffff80b8b81d70, option=3, send_timeout=0) at ipc_kmsg.c:1822 [opt]
    frame #11: 0xfffffff800a2b024f kernel.development`mach_msg_overwrite_trap(args=<unavailable>) at mach_msg.c:546 [opt]
    frame #12: 0xffffffff7f8e6d81d7 pasive_kernel_fuzz`trampline_mach_msg_overwrite_trap(args=0xffffffa756c8bf08) at mach_msg_overwrite_trap_trampline.c:131
    frame #13: 0xfffffff800a42cb09 kernel.development`mach_call_munger64(state=0xfffffff80ac13de20) at bsd_i386.c:573 [opt]
    frame #14: 0xfffffff800a25b466 kernel.development`hdl Mach_scall64 + 22
```

Root Cause

```
frame #0: 0xfffffff7f8d91e324 AMDFramebuffer`AMDFramebuffer::getPixelInformationFromTiming(AtiDetailedTimingInformation const&, IOPixelInformation*, int, int) + 388
AMDFramebuffer`AMDFramebuffer::getPixelInformationFromTiming:
-> 0xfffffff7f8d91e324 <+388>: movq (%rcx,%rdi,8), %rcx      — a)
  0xfffffff7f8d91e328 <+392>: movq %rsi, %rdi
  0xfffffff7f8d91e32b <+395>: movq %rcx, %rsi
  0xfffffff7f8d91e32e <+398>: callq 0xfffffff7f8ccbcfe0 ; Utilities::str_copy(char*, char const*, unsigned long)
(lldb) register read rcx
  rcx = 0xfffffff7f8d926030 AMDFramebuffer::getPixelInformationFromTiming(AtiDetailedTimingInformation const&, IOPixelInformation*, int, int)::PIXEL_ENCODINGS
(lldb) register read rdi
  rdi = 0xffffffffffff20000001
```

```
unsigned int v10; // [sp+D8h] [bp-28h]@1
int v11; // [sp+DCh] [bp-24h]@1
void *v12; // [sp+E0h] [bp-20h]@1
void *v13; // [sp+E8h] [bp-18h]@1
__int64 v14; // [sp+F0h] [bp-10h]@1
unsigned int v15; // [sp+FCh] [bp-4h]@2

v14 = a1;
v13 = a2;
v12 = a3;
v11 = a4;
v10 = a5;
v8 = 0;
bzero(a3, 0xACuLL);
if ( (signed int)v10 <= 2 )
{
```

use

```
text:0000000000021318
text:000000000002131C
text:0000000000021320
text:0000000000021324
text:0000000000021328
text:000000000002132B
text:000000000002132E
text:0000000000021333
text:0000000000021339
text:000000000002133C
text:0000000000021343
```

move the value(0xf2000001) of rbp+var_28 to rdi, and
extends it to 64 bits. then becomes 0xffffffffffff20000001
rdi*8 becomes so big , crash point

may be leak info if craft the value of rdi

however, if we craft the rdi value, the rdi*8 can be control by user

```
    mov    rsi, [rbp+var_28]
    add    rsi, 58h
    movsd rdi, [rbp+var_28]
    mov    rcx, [rcx+rdi*8]; unsigned int64
    mov    rdi, rsi      ; this
    mov    rsi, rcx      ; char *
    call   __ZN9Utilities8str_copyEPcPKcm ; Utilities::str_copy(char *,char const*,ulong)
    mov    r8d, 40h, `@`_
    mov    edx, r8d
    mov    rcx, 0xFFFFFFFFFFFFFFFh
    lea    rsi, __ZL15COMPONENT_MASKS ; COMPONENT_MASKS
    ...
```

Agenda

- ❖ Static Analysis for Kernel Extensions Attack Interfaces
- ❖ Enhanced PassiveFuzz
- ❖ Vulnerabilities Found
- ❖ Conclusion

Conclusion

- ❖ Introduce a method to analyze the attack surfaces of kernel extensions, then introduce an enhanced passive fuzz architecture on Apple system. Finally, we study one CVE case hunt by our fuzzer

Questions?

Contact US: @Lilang_Wu, @Flyic