Launch and Environment Constraints Deep Dive

OffSec

Csaba Fitzl
Twitter: @theevilbit

whoami

- lead content developer of "EXP-312: Advanced macOS Control Bypasses" @ OffSec
- ex red/blue teamer
- macOS bug hunter
- husband, father
- hiking, trail running 🥾 🏞️🏃

agenda

1. old macOS vulnerabilities
2. Launch Constraints (Ventura)
3. old and current third party vulnerabilities
4. Launch and Environment constraints (Sonoma)

old macOS vulnerabilities

TCC bypass with imagent.app

TCC bypass with imagent.app

- Found by Adam Chester (@_xpn_)
- imagent.app with TCC and keychain related entitlements
- loads plugins from:
  - imagent.app/Contents/PlugIns
- code signing allows 3rd party plugins
- copy app to /tmp/ and load your plugin

```
<key>com.apple.private.tcc.allow.overridable</key>
<array>
        <string>kTCCServiceAddressBook</string>
</array>

<key>keychain-access-groups</key>
<array>
        <string>ichat</string>
        <string>apple</string>
        <string>appleaccount</string>
        <string>InternetAccounts</string>
        <string>IMCore</string>
</array>
```

TCC bypass using Directory Utility.app, CVE-2020-27937

TCC bypass using Directory Utility.app, CVE-2020-27937

- found by Wojciech Regula (@_r3ggi)
- Directory Utility with admin rights to change user properties, like HOME
- allows plugins including non Apple
- copy app to /tmp/ and load our plugin
- change HOME -> new TCC.db -> our rules

```
<key>com.apple.private.tcc.allow</key>
<array>
    <string>kTCCServiceSystemPolicySysAdminFiles</string>
</array>
```

TCC bypass using configd, "powerdir"

TCC bypass using configd, "powerdir"

- Found by Jonathan Bar Or (@yo_yo_yo_jbo)
- configd has again user update rights (can change HOME)
- -b allows loading an bundle (including non Apple)
- normally launched by launchd but we could start it via command line as well

```
[Key] com.apple.private.tcc.allow
[Value]
    [Array]
        [String] kTCCServiceSystemPolicySysAdminFiles
```

Introducing Launch Constraints

Launch Constraints

- introduced in macOS Ventura (13)
- mitigates many logic vulnerabilities
- defines 3 constraints:
  - Self Constraints
  - Parent Constraints
  - Responsible Constraints

LC in Action

```
csaby@max /tmp % cp -r /System/Applications/FindMy.app .

csaby@max /tmp % open FindMy.app
The application cannot be opened for an unexpected reason, error=Error Domain=RBSRequestErrorDomain
Code=5 "Launch failed." UserInfo={NSLocalizedFailureReason=Launch failed.,
NSUnderlyingError=0x6000000032d0 {Error Domain=NSPOSIXErrorDomain Code=162 "Unknown error: 162"
UserInfo={NSLocalizedDescription=Launchd job spawn failed}}}
```

```
csaby@max /tmp % log stream | grep AMFI
2023-09-19 14:18:21.273482+0200 0x2e3486   Default     0x0                   0      0    kernel:
(AppleMobileFileIntegrity) AMFI: Launch Constraint Violation (enforcing), error info: c[1]p[1]m[1]e[2],
(Constraint not matched) launching proc[vc: 1 pid: 52468]: /private/tmp/FindMy.app/Contents/MacOS/
FindMy, launch type 0, failure proc [vc: 1 pid: 52468]: /private/tmp/FindMy.app/Contents/MacOS/FindMy
```

Launch Constraints Categories

LC Categories

- category = defines a set of launch constraints
- Ventura - 7 categories - documented by Linus Henze
- Sonoma - 18 categories - documented by Csaba Fitzl
- assigns each binary in the trust cache to a category

LC Category examples

```
Category 1:

    Self Constraint: (on-authorized-authapfs-volume || on-system-volume) && launch-type == 1 &&
validation-category == 1

    Parent Constraint: is-init-proc
```

- *on-authorized-authapfs-volume || on-system-volume* - System or Cryptex
- *launch-type == 1* - system service
- *validation-category == 1* - must present in the trust cache
- *is-init-proc* - launchd

```
/usr/libexec/routined
/usr/libexec/nehelper
/usr/libexec/remoted
/usr/libexec/seld
/usr/libexec/logd
/usr/libexec/thermalmonitord
```

LC Category examples

```
Category 2:

     Self Constraint: on-authorized-authapfs-volume || on-system-volume
```

- *on-authorized-authapfs-volume || on-system-volume* - System or Cryptex
- less restrictive

```
/usr/bin/brctl
/usr/bin/bputil
/usr/bin/bison
/usr/bin/bioutil
/usr/bin/binhex
/usr/bin/bc
/usr/bin/batch
```

trust cache

Trust Cache

- A few places:
  - /System/Library/Security/OSLaunchPolicyData
  - /System/Volumes/Preboot/[uuid]/boot/[long hex]/usr/standalone/firmware/FUD/BaseSystemTrustCache.img4
  - /System/Volumes/Preboot/[uuid]/boot/[long hex]/usr/standalone/firmware/FUD/StaticTrustCache.img4
- IMG4 and IM4P (P = Payload) files

Trust Cache

- IMG4 - extract IM4P
- IM4P - extract data
- pyimg4 Python utility

```
pyimg4 img4 extract -i BaseSystemTrustCache.img4 -p BaseSystemTrustCache.im4p
pyimg4 img4 extract -i StaticTrustCache.img4 -p StaticTrustCache.im4p
```

```
csaby@max /tmp % pyimg4 im4p info -i /System/Library/Security/OSLaunchPolicyData
Reading /System/Library/Security/OSLaunchPolicyData...
Image4 payload info:
  FourCC: ltrs
  Description: 1
  Data size: 329.14KB
  Encrypted: False

csaby@max /tmp % pyimg4 im4p extract -i /System/Library/Security/OSLaunchPolicyData -o
OSLaunchPolicyData.data
Reading /System/Library/Security/OSLaunchPolicyData...
Extracted Image4 payload data to: OSLaunchPolicyData.data
```

Trust Cache

- TC v2 can contain constraint category
- trustcache utility to analyze it
- category: 4th column

```
struct trust_cache_entry2 {
    uint8_t cdhash[CS_CDHASH_LEN];
    uint8_t hash_type;
    uint8_t flags;
    uint8_t constraintCategory;
    uint8_t reserved0;
} __attribute__((__packed__));
```

```
csaby@max /tmp % trustcache_macos_arm64 info OSLaunchPolicyData.data | head
version = 2
uuid = CCC03EBE-7949-460E-A335-14C6396FC927
entry count = 13713
000600f05b768de957b57afbd576ad031b7dc984 [none] [2] [2]
0008df4d1e0d4b276a82f3e086bea3e93670cf94 [none] [2] [2]
000c95ce2e4f99248a33e5c7f690452f89afc16b [none] [2] [0]
0019e93d101896746f77a3b7047d2d8281352fc5 [none] [2] [3]
0020f949545b505610f55f962520bfb4f1851f1d [none] [2] [2]
002655b46255b6fb2f5b2a4987345cc0e46836f8 [none] [2] [2]
002aa16545d098699c706c27a08e834243f5b984 [none] [2] [2]
```

reversing launch constraints

reversing LC

- defined in AMFI (AppleMobileFileIntegrity)
- download KDK to get the KEXT
- symbols with *kConstraintCategory** prefix
- extract symbols from KEXT

```
kConstraintCategory1_Self
kConstraintCategory2_Self
kConstraintCategory3_Self
kConstraintCategory4_Self
kConstraintCategory5_Self
kConstraintCategory6_Self
kConstraintCategory7_Self
kConstraintCategory8_Self
kConstraintCategory9_Self
kConstraintCategory10_Self
kConstraintCategory12_Self
kConstraintCategory13_Self
kConstraintCategory14_Self
kConstraintCategory15_Self
kConstraintCategory16_Self
kConstraintCategory17_Self
kConstraintCategory18_Self
kConstraintCategory20_Self
kConstraintCategory1_Parent
kConstraintCategory4_Parent
kConstraintCategory5_Parent
kConstraintCategory6_Parent
kConstraintCategory8_Parent
kConstraintCategory14_Parent
kConstraintCategory15_Parent
kConstraintCategory16_Parent
kConstraintCategory18_Parent
```

7075020101B07030420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF30100C0B6C61756E63682D74797065020101301800C1376616C69646174696F6E2D63617465676F727902010170490201018044304200C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF708183020101B07E30420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF301E0C0B6C61756E63682D74797065B00F300D0C0324696E3006020100020101301800C1376616C69646174696F6E2D63617465676F727902010170818302010 1B07E30420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF301E0C0B6C61756E63682D74797065B00F300D0C0324696E3006020100020101301800C1376616C69646174696F6E2D63617465676F727902010170818302010 1B01A30180C1376616C69646174696F6E2D63617465676F727902010170818 3B2020101B081AC307E0C03246F72B07730230C1E696E2D74632D776974682D636F6E7374726 1696E742D63617465676F727901010030150C1069732D7369702D70726F746563746564010 1FF30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF30100C0B6C61756E63682D74797065020101301800C1376616C69646174696F6E2D63617465676F727902010170750201018 07030420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF30100C0B6C61756E63682D74797065023018 0C1376616C69646174696F6E2D63617465676F727902010170819902010 1B0819330420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF30130C0E6170706C652D696E7465726E616C0101FF301E0C0B6C61756E63682D74797065B00F300D0C0324696E3006020100020102301 80C1376616C69646174696F6E2D63617465676F727902010170818302010 1B07E30420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF301E0C0B6C61756E63682D74797065B00F300D0C0324696E3006020100020102301 80C1376616C69646174696F6E2D63617465676F727902010170818702010 1B0818130420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF30210C0B6C61756E63682D74797065B01230100C0324696E3009020100020101020102301 80C1376616C69646174696F6E2D63617465676F727902010170 3F020101B03A301E0C0B6C61756E63682D74797065B00F300D0C0324696E3006020100020102301 80C1376616C69646174696F6E2D63617465676F727902010170750201018 07030420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF30100C0B6C61756E63682D74797065023301 80C1376616C69646174696F6E2D63617465676F727902010170818 3B7020101B081B130819C0C03246F72B08194307D0C0424616E64B07530590C03246F72B05230150C1069732D7369702D70726F746563746564010 1FF30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF30180C1376616C69646174696F6E2D63617465676F727902010130130C0E6170706C652D696E7465726E616C0101FF30100C0B6C61756E63682D74797065027078020101B07330420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797737465 6D2D766F6C756D650101FF30130C0E646576656C6F7065722D6D6F64650101FF30180C1376616C69646174696F6E2D63617465676F727902010170818702010 1B0818130420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF30210C0B6C61756E63682D74797065B01230100C0324696E3009020100020101020102301 80C1376616C69646174696F6E2D63617465676F727902010170180201018 01330110C0C69732D696E69742D70726F630101FF70818A020101B08184308181 0C03246F72B07A30650C0424616E64B05D30150C106F6E2D73797374656D2D766F6C756D650101FF302A0C127369676E696E672D6964656E746966965720C14636F6D2E6170706C652E6D62666C6F6167656E7430180C1376616C69646174696F6E2D63617465676F727902010130110C0C69732D696E69742D70726F630101FF70818A020101B08184308181 0C03246F72B07A30650C0424616E64B05D30150C106F6E2D73797374656D2D766F6C756D650101FF302A0C127369676E696E672D6964656E746966965720C14636F6D2E6170706C652E6D62666C6F6167656E7430180C1376616C69646174696F6E2D63617465676F727902010130110C0C69732D696E69742D70726F630101FF70819A020101B08194308191 0C03246F72B0818930740C0424616E64B06C30130C0E6170706C652D696E7465726E616C0101FF30550C0C656E7469746C656D656E7473B04530430C0624717565727930393 02F0201010C2A636F6D2E6170706C652E707269766174652E7365742D6C61756E63682D747970652E696E7465726E616C300602010A02010130110C0C69732D696E69742D70726F630101FF7081B4020101B081AE30420C03246F72B03B30220C1D6F6E2D617574686F72697A65642D6175746861706732D766F6C756D650101FF30150C106F6E2D73797374656D2D766F6C756D650101FF304E0C127369676E696E672D6964656E746966696572B03830360C0324696E302F0C15636F6D2E6170706C652E737973646961676E6F73650C16636F6D2E6170706C652E737973646961676E6F7365643018 0C1376616C69646174696F6E2D63617465676F727902010170180201018 01330110C0C69732D696E69742D70726F630101FF7018020101B01330110C0C69732D696E69742D70726F630101FF7018020101B01330110C0C69732D696E69742D70726F630101FF7081A0020101B0819A3081970C03246F72B0818F30780C0424616E64B07030540C03246F72B04D30110C0C69732D696E69742D70726F630101FF30380C127369676E696E672D6964656E746966965720C22636F6D2E6170706C652E436F726544657669636652E6474646562756770726F7879643018 0C1376616C69646174696F6E2D63617465676F727902010130130C0E6170706C652D696E7465726E616C0101FF

reversing LC

- ASN.1 DER encoded data (serialized)
- many tools which can decode it, like python-asn1
- DER encoding:
  - Type
  - Length
  - Value

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201301800C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D6175746861706732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D74797065020101301800C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170667 32D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201301 80C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201301800C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201013018 0C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B0703042 0C 03 246F72 B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170667 32D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201013018 0C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D6175746861706732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201301 80C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D6175746861706732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201301800C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
3022**0C1D6F6E2D617574686F72697A65**
**642D6175746861706732D766F6C756D**
**65**0101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201301800C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D6175746861706732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201301800C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170666732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201301800C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201301800C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170666732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201301800C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170666732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201013018 0C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170666732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D74797065020101301800C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D74797065020101301 80C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201013018 0C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201013018 0C
13 76616C69646174696F6E2D63617465
676F7279 020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170667 32D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D747970650201 30180C
1376616C69646174696F6E2D63617465
676F727902 01 01
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

DER decoding example

```
7075020101B07030420C03246F72B03B
30220C1D6F6E2D617574686F72697A65
642D617574686170732D766F6C756D
650101FF30150C106F6E2D7379737465
6D2D766F6C756D650101FF30100C0B6C
61756E63682D74797065020101301 80C
1376616C69646174696F6E2D63617465
676F7279020101
```

```
[A] SEQUENCE
  [U] INTEGER: 1
  [C] SEQUENCE
    [U] SEQUENCE
      [U] UTF8STRING: $or
      [C] SEQUENCE
        [U] SEQUENCE
          [U] UTF8STRING: on-authorized-authapfs-volume
          [U] BOOLEAN: True
        [U] SEQUENCE
          [U] UTF8STRING: on-system-volume
          [U] BOOLEAN: True
    [U] SEQUENCE
      [U] UTF8STRING: launch-type
      [U] INTEGER: 1
    [U] SEQUENCE
      [U] UTF8STRING: validation-category
      [U] INTEGER: 1
```

(on-authorized-authapfs-volume || on-system-volume) && launch-type == 1 && validation-category == 1

attack mitigation

LC attack mitigation

- imagent.app & Directory Utility.app
  - (on-authorized-authapfs-volume || on-system-volume)
  - wouldn't be able to start a copy

- configd
  - Parent Constraint: is-init-proc + system service
  - wouldn't be able to start from command line

typical third party attacks

XPC daemon attacks

- many blog post series from Wojciech Regula, Csaba Fitzl and others
- XPC services which run as root are dangerous
- XPC service must ensure that only the real client can connect
- client validation is ~~hard~~ not straightforward
  - must use audit token
  - client must be signed by developer certificate issued by Apple
  - client's version must be checked and / or
  - client shouldn't posses dangerous entitlements (e.g.: com.apple.security.cs.disable-library-validation)

Embedded XPC attacks

- not aware of any in third party
- rare even with Apple binaries
  - https://xlab.tencent.com/en/2021/01/11/cve-2020-9971-abusing-xpc-service-to-elevate-privilege/
  - https://jhftss.github.io/CVE-2022-26712-The-POC-For-SIP-Bypass-Is-Even-Tweetable/

Electron framework attacks

- Electron based apps are popular (Chrome, Signal, Slack, Discord, MS Teams, ...)
- often have TCC permissions (Camera, Microphone, ...)
- subject to injection attacks (local attack are not in their threat model)
- ELECTRON_RUN_AS_NODE - allows node.js cli interaction
  - can be disabled
- arguments: --inspect and --remote-debugging-port - allow debugger attachment
  - --remote-debugging-port can't be disabled

dylib injection

- if "com.apple.security.cs.disable-library-validation" present
- often used if third party plugins must be supported
- opens up the attack surface for XPC and TCC bypasses
  - -> can inject code into the clients

Launch and Environment Constraints (for 3rd parties)

“In this talk I will talk about two mitigations which Apple introduced in order to protect against many types of logic vulnerabilities. Launch Constraints was introduced in macOS Ventura, and they can control who can launch a built-in system application and how. Environment Constraints were introduced in Sonoma, and it's basically the extension of Launch Constraints for third party apps. These two features are probably the most impactful when it comes to exploitation. I will review them in detail, how they are set up, what they do exactly, and what kind of vulnerability classes they mitigate. I will also go through a couple of past vulnerabilities, which could not have been exploited with these constraints present. Finally I will walk through how various third party apps should be set up in order to be secure.”

–Csaba Fitzl

"In this talk I will talk about two mitigations which Apple introduced in order to protect against many types of logic vulnerabilities. Launch Constraints was introduced in macOS Ventura, and they can control who can launch a built-in system application and how. Environment Constraints were introduced in Sonoma, and it's basically the extension of Launch Constraints for third party apps. These two features are probably the most impactful when it comes to exploitation. I will review them in detail, how they are set up, what they do exactly, and what kind of vulnerability classes they mitigate. I will also go through a couple of past vulnerabilities, which could not have been exploited with these constraints present. ***Finally I will walk through how various third party apps should be set up in order to be secure."***

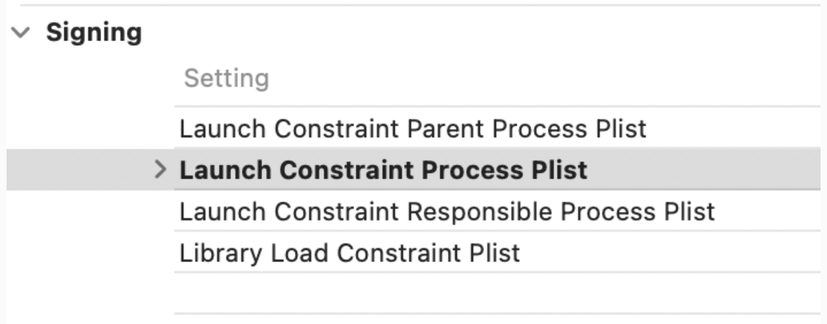
–Csaba Fitzl

“*In this talk I will talk about two mitigations which Apple introduced in order to protect against many types of logic vulnerabilities. Launch Constraints was introduced in macOS Ventura, and they can control who can launch a built-in system application and how. Environment Constraints were introduced in Sonoma, and it's basically the extension of Launch Constraints for third party apps. These two features are probably the most impactful when it comes to exploitation. I will review them in detail, how they are set up, what they do exactly, and what kind of vulnerability classes they mitigate. I will also go through a couple of past vulnerabilities, which could not have been exploited with these constraints present.* ***Finally I will ~~walk through how various third party apps should be set up in order to be secure.~~ rant a little about why it doesn't solve any of the real issues***”

–Csaba Fitzl

intro

- now LC available for 3rd party apps
- can define all 3 constraints (self, parent, responsible)
- additionally +1 library load constraint
- defined in code requirement in Xcode
- well documented in Apple Developer Documentation



XPC protection - embedded

- likely* works
  - *needs testing

*"Now let's walk through some process relationships and talk about how you can use launch constraints to secure them. First assume that MyDemo.app is your app. You can set a self constraint on my MyDemo.app to require that it launch as an application from Launch Services. When your app requests a connection to your XPC service, launchd spawns the XPC service and is the parent of that XPC service but your app is "responsible" for that XPC service. You could set a responsible process constraint on MyXPCDemo.xpc to indicate that only MyDemo.app should be responsible for it." **

* *WWDC2023: Protect your Mac app with environment constraints*

XPC "protection" - daemon

- sounds like LC makes XPC secure
- responsible process: XPC service itself, not the client (FB13206884)
  - maybe ok (?)
  - is resp. proc meaningful?
- if the service is already running then LC is not in play

*"Now let's walk through some process relationships and talk about how you can use launch constraints to secure them. First assume that MyDemo.app is your app. You can set a self constraint on my MyDemo.app to require that it launch as an application from Launch Services. When your app requests a connection to your XPC service, launchd spawns the XPC service and is the parent of that XPC service but your app is "responsible" for that XPC service. You could set a responsible process constraint on MyXPCDemo.xpc to indicate that only MyDemo.app should be responsible for it." **

** WWDC2023: Protect your Mac app with environment constraints*

"Securing" Electron applications

- posix_spawn is dangerous
- launch type = 3 = launch as application
- but!!!
- but!!!
- open command launches as application
  - can pass env vars and arguments

*"Just like in real parent-child relationships, parent processes have a huge amount of influence over how a child behaves. On macOS, the power to posix_spawn another process gives the parent the ability to control nearly all input to the child. The parent process can also limit the child's access to system resources. This level of control can cause the child to load unexpected code, to run unexpected features, or to behave in ways that make the process more vulnerable to attack." **

** WWDC2023: Protect your Mac app with environment constraints*

"Securing" Electron applications

or just use the API

```
NSWorkspaceOpenConfiguration *conf = [NSWorkspaceOpenConfiguration configuration];
conf.environment = @{
    @"ELECTRON_RUN_AS_NODE": @"1"
};
[[NSWorkspace sharedWorkspace] openURL:[NSURL fileURLWithPath:@"/Applications/Electron.app"]
                               configuration:conf
                               completionHandler:nil];
```

library load constraints

- if you used "com.apple.security.cs.disable-library-validation"
- solves third party plugin support issues
- problem:
  - assumes you are aware of all the plugins

conclusion

conclusion

- LC for Apple binaries
  - great improvement
  - mitigates many common attacks
- LC for 3rd parties
  - there is potential, but needs improvement
  - currently doesn't impact the most common attacks
  - library load constraints can be useful
  - start to use it

OffSec™

Csaba Fitzl
Twitter: @theevilbit

Resources

- flaticon.com - Freepik, rsetiawan