

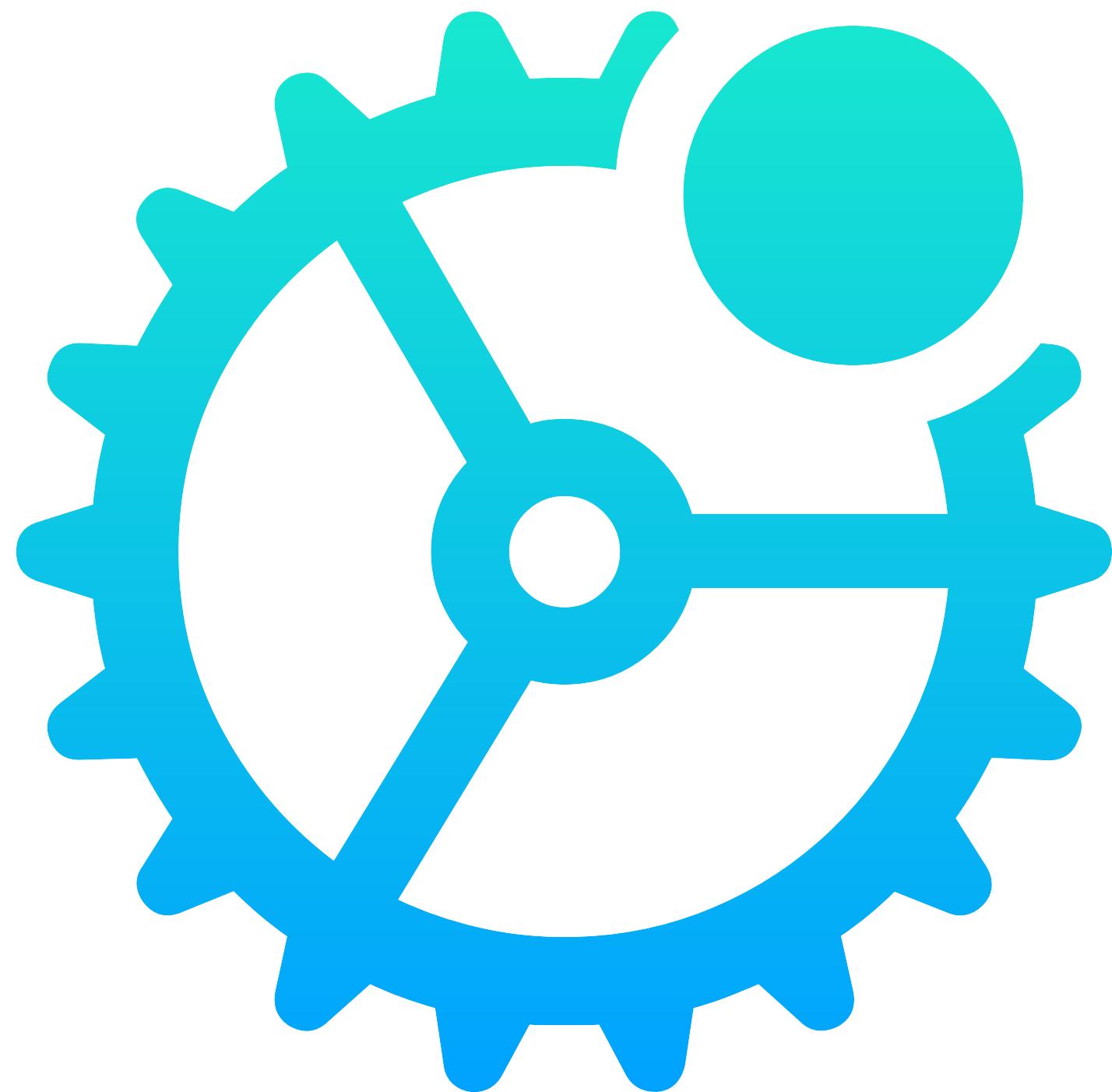
# Apple's (not so)<sup>1</sup> Rapid Security Response

Objective by the Sea v7.0

Mykola Grymalyuk - December 6th, 2024

# Bill of Materials

- What is a Rapid Security Response?
- What gaps does it fill?
- What is the underlying technology?
- How do they work?
- Where do they fall short?
- Why are they no longer used?
- Where might they be secretly hiding today?



# \$ '/usr/bin/whoami'



> "Mykola Grymalyuk"

- Lead Security and Software Engineer at RIPEDA Consulting.
- CVEs with Apple, Jamf, VMware, etc
- Project lead of OpenCore Legacy Patcher.
- Breaking macOS internals on [khronokernel.com](http://khronokernel.com).



**What is a Rapid Security  
Response?**

# What is a Rapid Security Response?

A security update that comes out faster than other security updates...

# What is a Rapid Security Response?

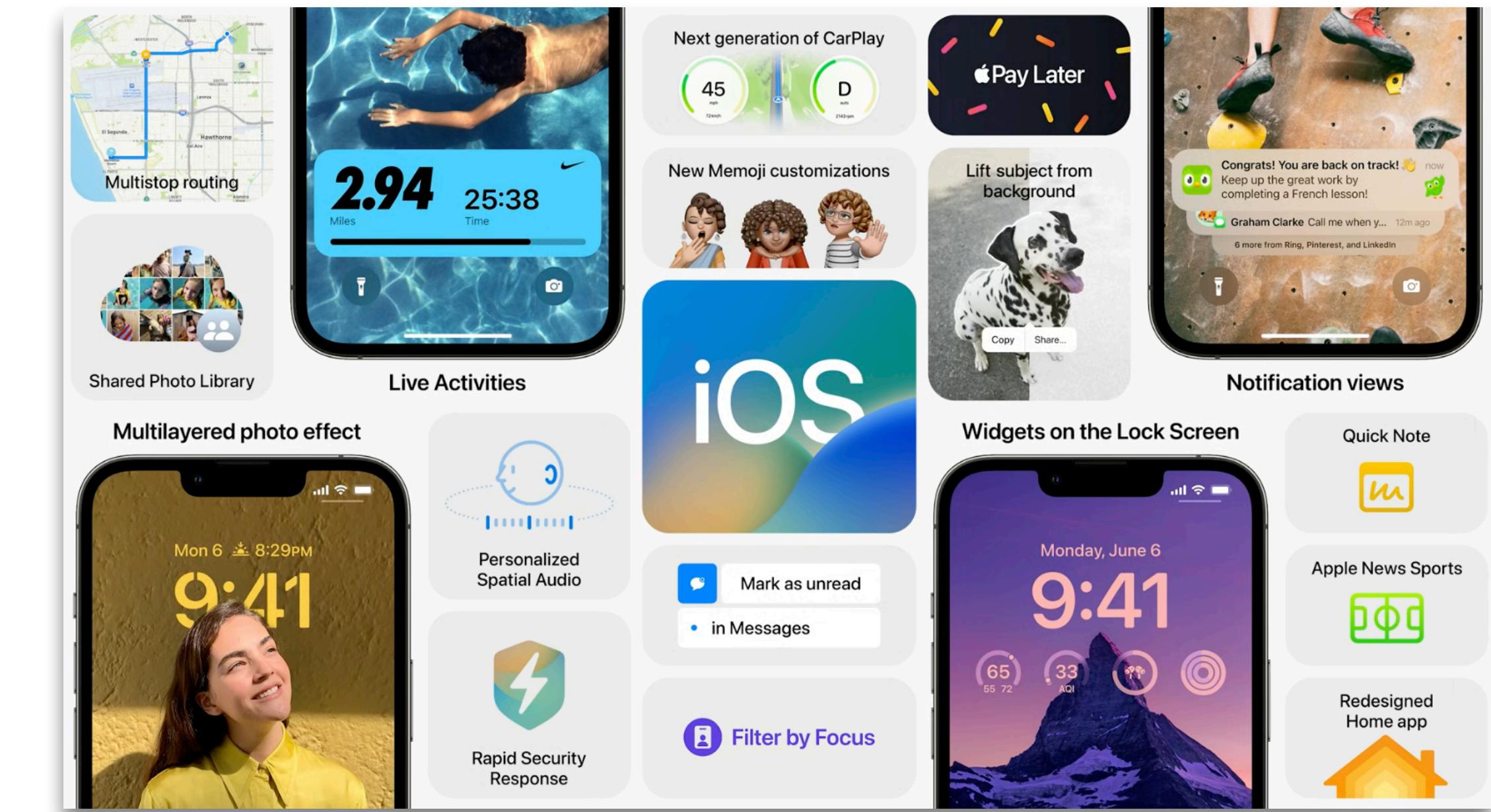
A security update  comes out faster than other security updates...

# Rapid Security Response

WWDC 2022



macOS 13, Ventura

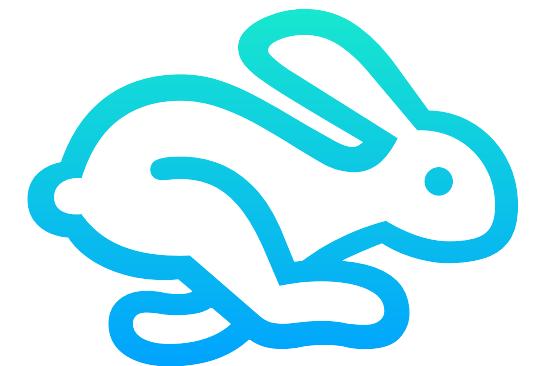


iOS 16

**What gaps does it fill?**

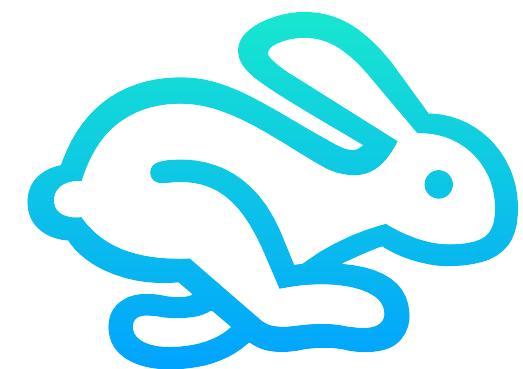
# What gaps does it fill?

# What gaps does it fill?



1. Quickly handle zero days

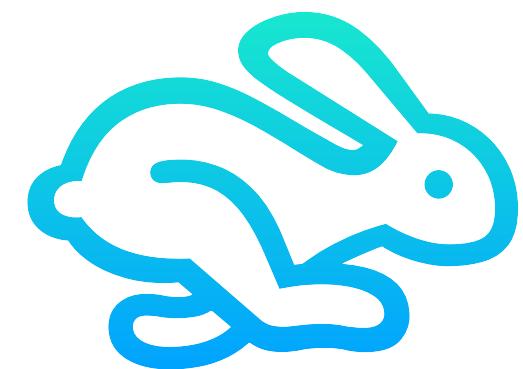
# What gaps does it fill?



1. Quickly handle  
zero days

2. Reduce update times  
and reboots required

# What gaps does it fill?



1. Quickly handle  
zero days

2. Reduce update times  
and reboots required

3. Ability to revert  
updates

**RSR Versioning: X.Y.Z (a,b,c)**

# **RSR Versioning: X.Y.Z (a,b,c)**

**macOS 15.1.1**

**iOS 18.1.1**

# **RSR Versioning: X.Y.Z (a,b,c)**

**macOS 15.1.1 (a)**

**iOS 18.1.1 (a)**

# RSR Versioning: X.Y.Z (a,b,c)

macOS 15.1.1 (a)

iOS 18.1.1 (a)

Host: /System/Library/CoreServices/SystemVersion.plist

RSR: /System/Volumes/Preboot/Cryptexes/OS/System/Library/CoreServices/SystemVersion.plist

# RSR Versioning: X.Y.Z (a,b,c)

macOS 15.1.1 (a)

iOS 18.1.1 (a)

Host: /System/Library/CoreServices/SystemVersion.plist

24B91

RSR: /System/Volumes/Preboot/Cryptexes/OS/System/Library/CoreServices/SystemVersion.plist

24B880910b

**What is the underlying  
technology?**

**What is the underlying  
technology?**

**Let's roll back to iOS 14**

# Apple Security Research Device Program

# Apple Security Research Device Program

- Released in early 2021
- Launched with iOS 14 and an Apple provided iPhone 11 (SRD)
- Uses a new “**Cryptex**” system for research and development



A black smartphone is shown from a top-down perspective, displaying a terminal window with a black background and white text. The text is a detailed log of a boot process, showing various kernel messages, device detections, and configuration steps. Key terms visible in the log include "AppleEmbedded", "NVMeController", "APL", "APFS", "APFS\_XART", and "spaceman". The log ends with a "nobrowse" command at the bottom.

```
..._005453 wlan0.A1
...ual boot AppleEmbedded
STYPE - 2
...NS2:client1 nvme_admin...
Virtual boot AppleEmbeddedNVMeController::AllocatedNodes(bool)::1588:Creating blockdevice with NSID -3
NSTYPE - 3
Virtual boot AppleEmbeddedNVMeController::AllocatedNodes(bool)::1588:Creating blockdevice with NSID - (
NSTYPE - 6
[NS2:client1] nvme_admin_handle_identify:namespace (6)
[effaceable:INIT] found current generation, 329, in group 0
[effaceable:INIT] started
Virtual boot AppleEmbeddedNVMeController::AllocatedNodes(bool)::1588:Creating blockdevice with NSID - 1
NSTYPE - 8
[NS2:client1] nvme_admin_handle_identify:namespace (7)
Virtual boot AppleEmbeddedNVMeController::StartController::1590:Setting NAND status to Ready
[NS2:client1] nvme_admin_set_nvme_and_event_controller_and_event_setter::Setting platform debug Flags to 0x0000000000000001
000012.01068 wlan0.A[14] initWlanTimerAndEventSetterAndModuleInstance@156: CCFlags: 0x0. CLevel: 127
ConsoleFlags: 1
000012.057163 wlan0.A[15] startChipUART0283: Starting...
000012.059112 wlan0.A[16] startChipUART0393: Workloop Enabled
000012.060070 wlan0.A[17] setChipEventLogFlags@28584: Setting Chip evene fChipEventLogFlags to 0x0
000012.067713 wlan0.A[18] getModuleInfo@4014: fMan s#B1 fProd M#GDF m#4.5 V#u
000012.071003 wlan0.A[19] start#2035:Checking firmware loaded@0
Got boot device = IOService://AppleARMPE/arm-top@0F000000/appleT803xI0/ans#077400000/AppleASCRawP2/lop-ar
s-nub/RTBuddyV2/RTBuddyService/AppleAN52CGNVMeController@NS_01@IOBlockerStorageDriver/APPLE SSD AP00
Media/IOGPUIPartitionScheme/Container@1
BSD root: disk0s1, major: 1, minor: 1
apfs_vfsop_mountroot:1865: apfs: mountroot called!
dev_init:297 disk0s1 device accelerated crypto: 3 (compiled @ May 17 2020 19:16:44)
dev_init:300 disk0s1 device_handle block size 4096 block count 15624989 features 22 internal solidstate
nx_kernel_mount: 1134: disk0s1 initializing ccache w/hash_size 4096 and cache size 10064
nx_kernel_mount: 1224: disk0s1 container cleanly-unmounted flag set.
AppleConvergedPCIICEB8::powerStateWillChange: device 0x1052daa49, stateNumber 1
nx_kernel_mount: 1402: disk0s1 checkpoint search target xid 933. best xid 933 # 185
apfs_vfsop_mount:1666: Preempting the current thread for disk0s1
handle_snapshot_mount@13: mounting snapshot w/snap_xid 58 and sblock old 0xecbb9f
handle_snapshot_mount@868: setting dev block size to 4096 from $12
handle_mount@157: vol-uuid: 3FFFB50C-5949-460F-A394-E8SACDE52287 block size: 4096 block count: 15624985
(unencrypted: flags: 0x1; features: 28.0.12)
nx_volume_group_update@6545: Volume com.apple.os.update=1479F0C480065208088AE983A99323123C276E75FA5B00
06F04B8E58B4F77FE75A8C86E3078BD00272094EB1D52FE23 is not in a volume group
apfs_vfsop_mount:1852: disk0s1s1:0 mounted volume: System
dyld: setting com.page 0x0
AppleConvergedPCIICEB8::openServiceGated:
Darwin Bootstrapper Version 7.0.0: Mon May 18 02:30:43 PDT 2020; root:/libxpc_executables-1983-131/launc
hd/RELEASE_ARM64
boot_args = -v
AppleConvergedPCIICEB8::openServiceGated: pci device open ret 1
Thu May 21 20:46:06 2020 localhost com.apple.xpc.launchd[1] (com.apple.xpc.launchd.domain.system) <Not
ce>: entering endemand node
** Checking the container superblock.
** Checking the object map.
** Checking the AFPS volume superblock.
[AP:C535L27iPhone11.cpp:422] cond "pcmResource->getBufferSize() > 0" fail. _handleLoadMemoryPlaybackR
source.
[AP:C535L27iPhone11.cpp:539] Failed to load MPB (State=1)
** Checking the APFS volume superblock.
** The volume System was formatted by newfs_apfs (945.200.129.100.8) and last modified by apfs_kext (1
29.0.0.0.2).
** Checking volume.
** Checking the APFS volume superblock.
** The volume Data was formatted by newfs_apfs (1629.0.0.0.2) and last modified by apfs_kext (1629.0.
0.2).
** Checking volume.
** Checking the APFS volume superblock.
** The volume XART was formatted by newfs_apfs (1629.0.0.0.2) and last modified by apfs_kext (1629.0.
0.2).
** Checking volume.
** Checking the APFS volume superblock.
** The volume Baseband Data was formatted by newfs_apfs (1629.0.0.0.2) and last modified by apfs_kext (1
629.0.0.0.2).
** Checking volume.
** Checking the APFS volume superblock.
** The volume Hardware was formatted by newfs_apfs (1629.0.0.0.2) and last modified by apfs_kext (1629.
0.0.2).
** Checking volume.
** Checking the APFS volume superblock.
** The volume Preboot was formatted by newfs_apfs (1629.0.0.0.2) and last modified by apfs_kext (1629.
0.0.0.2).
** Checking volume.
** Checking the APFS volume superblock.
** The volume Update was formatted by newfs_apfs (1629.0.0.0.2) and last modified by apfs_kext (1629.0.
0.0.2).
** QUICKCHECK ONLY: FILESYSTEM CLEAN
mount: found boot container: /dev/disk0s1, data volume: /dev/disk0s1s2
handle_mount@157: vol-uuid: 81318F8-4652-4A00-A7CD-716A95122C91 block size: 4096 block count: 15624981
(unencrypted: flags: 0x1; features: 8.0.2)
handle_mount@50: setting dev block size to 4096 from $12
nx_volume_group_update@6539: Volume Preboot role 10 Not a System or data volume
apfs_vfsop_mount:1852: disk0s1s1:0 mounted volume: Preboot
/dev/disk0s1s1 on /private/preboot (apfs, local, nodev, dosuid, read-only, journaled, noatime, nobrowse
)
spaceman_metazone_init@230: disk0s1 metazone for device 0 of size 488280 blocks (encrypted: 15136709-1:
380049 unencrypted: 15380849-15624989)
spaceman_datazone_init@491: disk0s1 allocation zone on dev 0 for allocations of 1 blocks starting at pi
ddr 4096000
spaceman_datazone_init@491: disk0s1 allocation zone on dev 0 for allocations of 2 blocks starting at pi
ddr 32768
spaceman_datazone_init@491: disk0s1 allocation zone on dev 0 for allocations of 3 blocks starting at pi
ddr 65536
spaceman_datazone_init@491: disk0s1 allocation zone on dev 0 for allocations of 1 blocks starting at pi
ddr 98304
dev_dump@256: Aggregate constructed: dev=<ptr> d1=0 dv_num_slice=27 dv_num_slice_blk=589824 dv_num_lsl=
ce_blk=289565
handle_mount@517 vol-uuid: F6CC2E80-AD8E-42AA-9593-BA278652A942 block size: 4096 block count: 15624989
(unencrypted: flags: 0x1; features: 8.0.2)
handle_mount@530: setting dev block size to 4096 from $12
nx_volume_group_update@6539: Volume xART role 100 Not a System or data volume
`->apfs_vfsop_mount:1852: disk0s1s1:0 mounted volume: xART
`->disk0s1s1 on /private/xarts (apfs, local, nodev, nosuid, journaled, noatime, nobrowse)
```

# How do RSRs/Cryptexes work?

# How do RSRs/Cryptexes work?



Preboot Volume

The screenshot shows a Mac OS X Finder window titled "Preboot". The sidebar on the left lists "Favorites" (Recents, Applications, Desktop, Documents, Downloads) and "Locations" (iCloud Drive). Below these are "Tags": Red, Orange, Yellow, Green, Blue, Purple, Gray, and All Tags... A warning message at the top of the sidebar reads: "Internal Hard Disk. Handle the hard drive carefully to avoid damaging the circuit board. Make sure you are properly grounded." The main pane displays a file tree under the path "Preboot > 16FB6020-F25D-485D-9807-1DEE6E1CEF59 > cryptex1 > current". The "current" folder contains two large disk images: "app.dmg" (23.1 MB) and "os.dmg" (5.65 GB), both of which are selected and highlighted in blue. Other files visible include "app.dmg.root\_hash", "app.dmg.trustcache", "apicket.vma2macosap.B041951382064679.im4m", "apicket.vma2macosap.im4m", "BuildManifest.plist", "os.clone.dmg", "os.dmg.root\_hash", "os.dmg.trustcache", "RestoreVersion.plist", "SystemVersion.plist", and several folders like "etc", "Library", "restore", "System", "usr", "var", "com.apple.security.cryptexd", "Cryptexes", and "staging". The status bar at the bottom shows the full path: "Preboot > 16FB6020-F25D-485D-9807-1DEE6E1CEF59 > cryptex1 > current".

/System/Volumes/Preboot/{OS UUID}/cryptex1/current

# How do RSRs/Cryptexes work?



## OS.dmg

Frameworks & dyld\_shared\_cache

Grafted to /System/Cryptexes/OS

## Cryptexes

Mounted via APFS.kext

Validated by root\_hash & includes trushcache



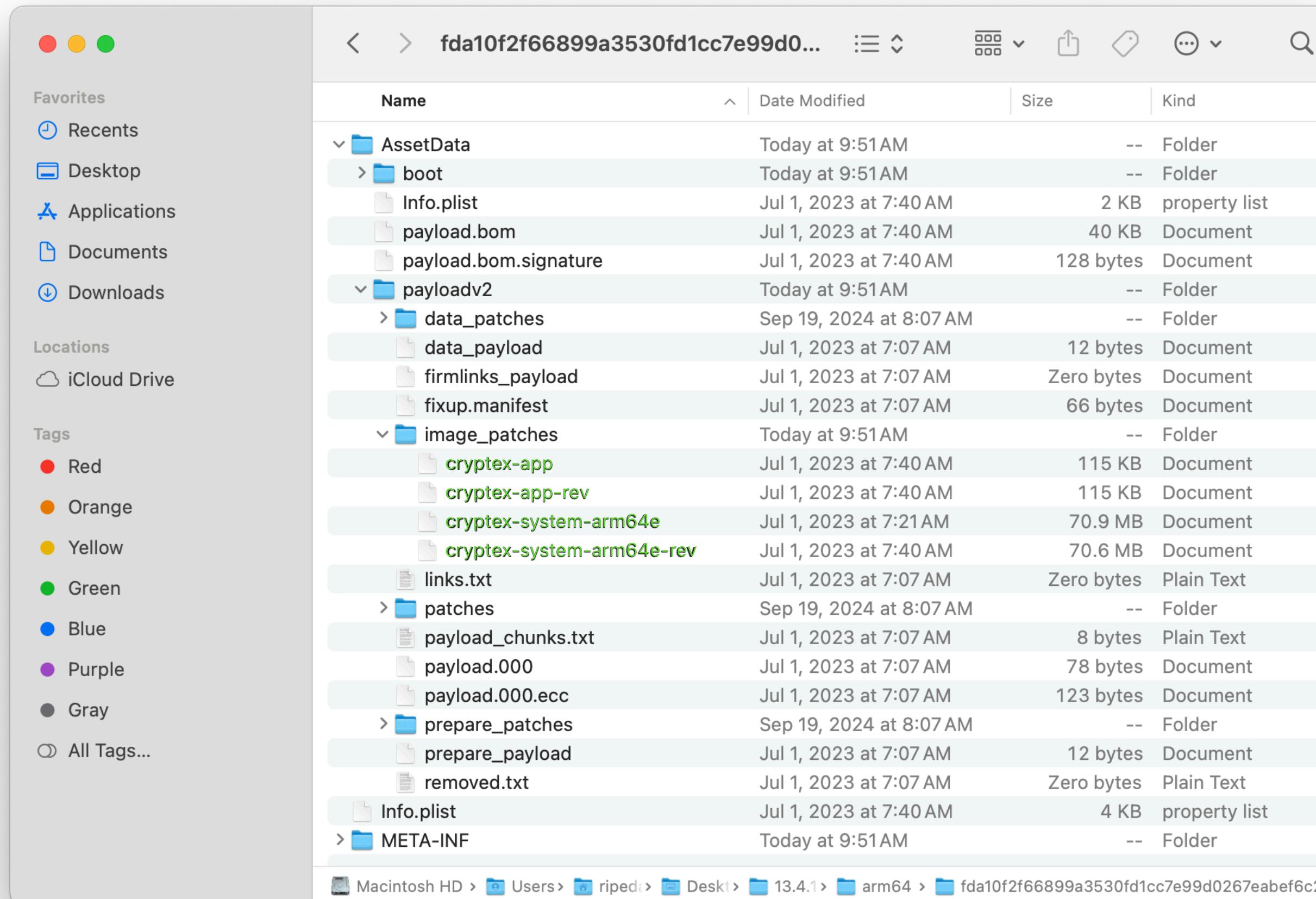
## App.dmg

Safari & Co.

Grafted to /System/Cryptexes/App

RSRs applied via RIDIFF10

# How do RSRs/Cryptexes work?



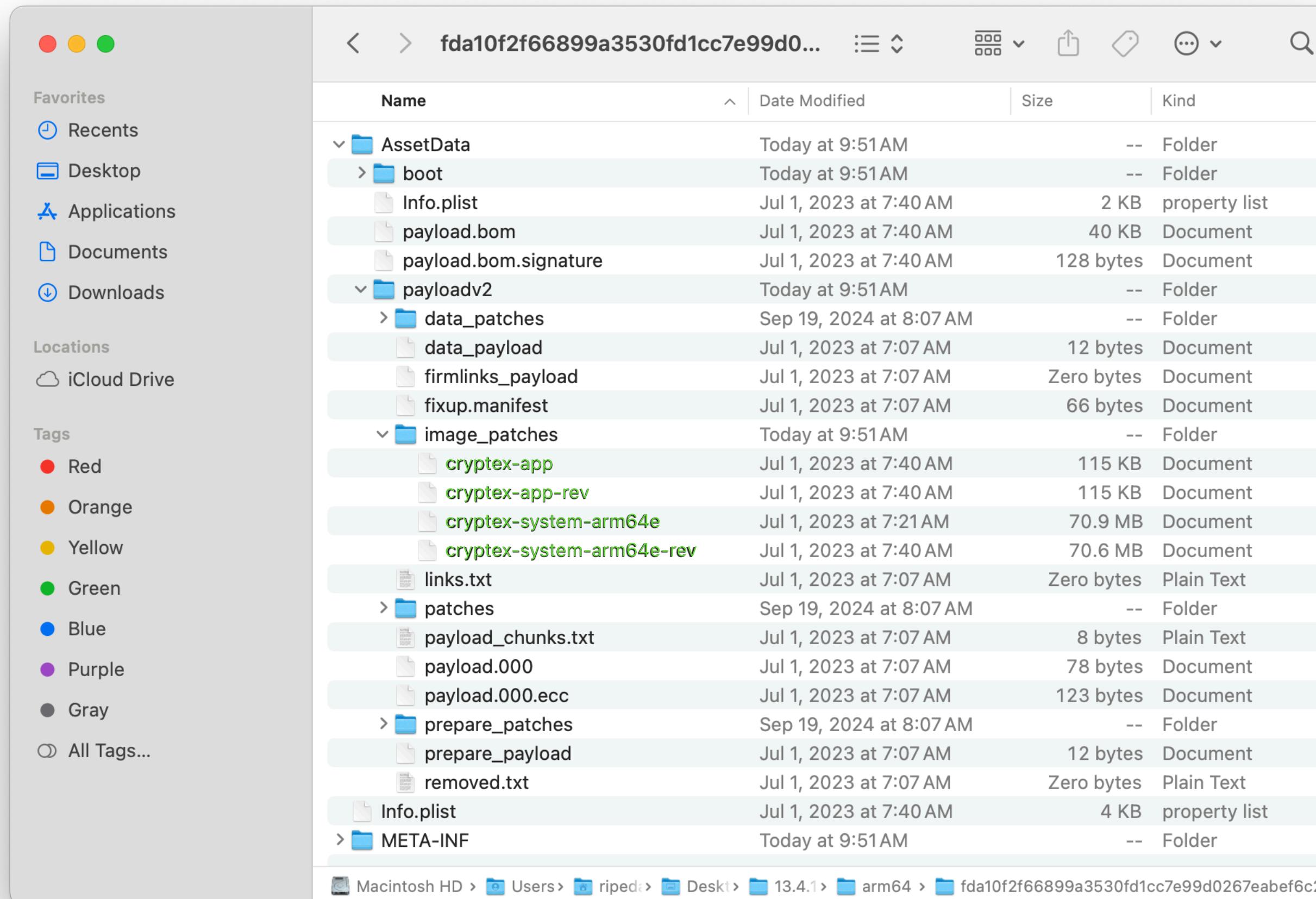
**cryptex-app**

**cryptex-app-rev**

**cryptex-system-{arch}**

**cryptex-system-{arch}-rev**

# How do RSRs/Cryptexes work?



**cryptex-app**

**cryptex-app-rev**

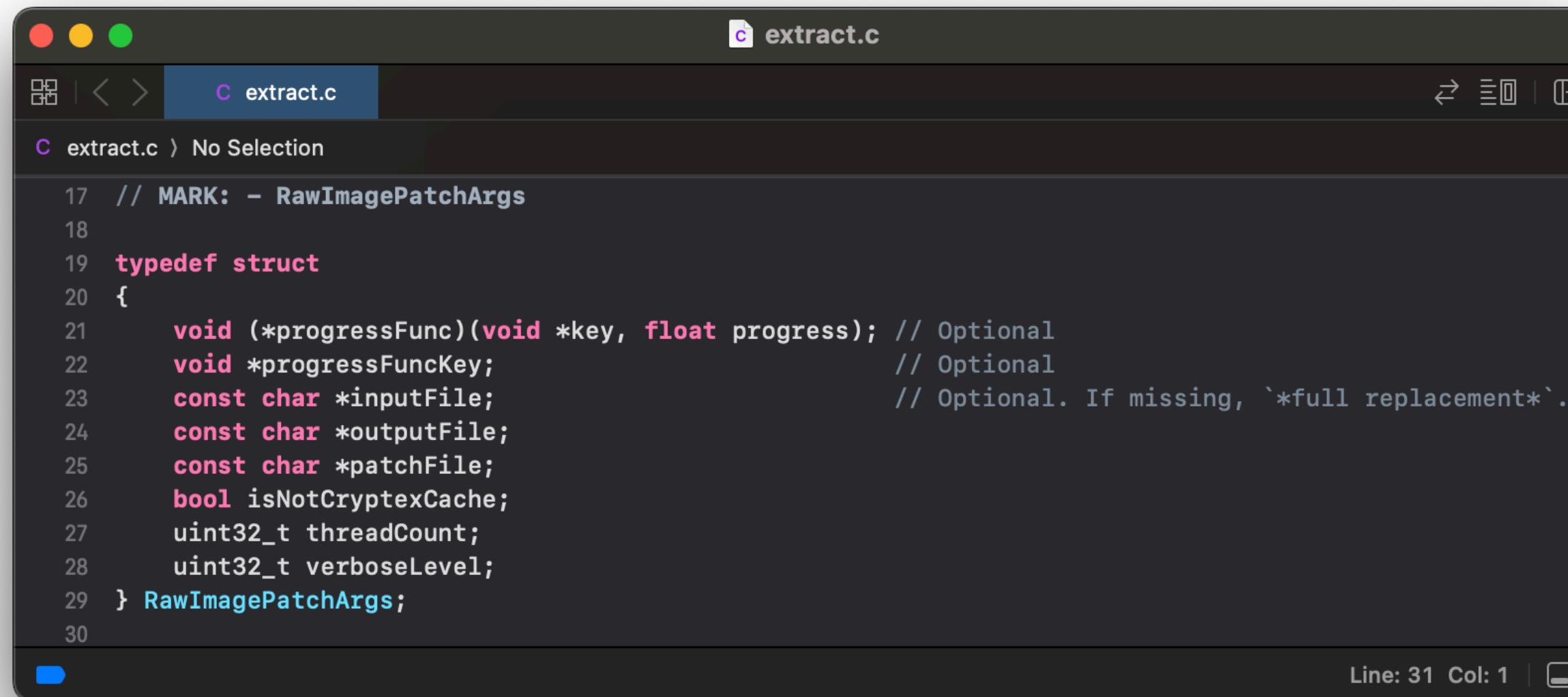
**cryptex-system-{arch}**

**cryptex-system-{arch}-rev**

**RSRs can't be applied on top each other**

# How do RSRs/Cryptexes work?

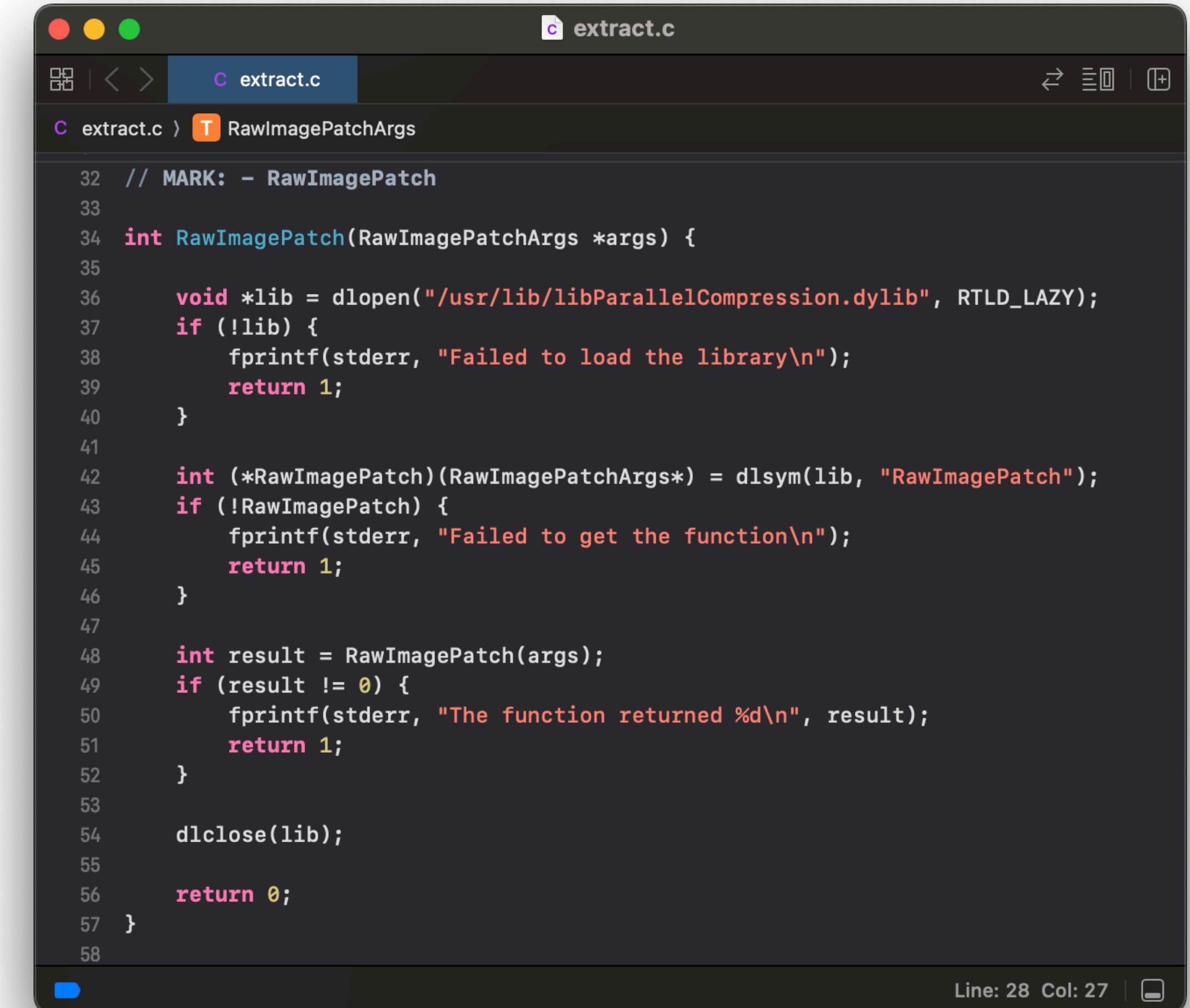
## /usr/lib/libParallelCompression.dylib's RawImagePatch()



```
extract.c
C extract.c > No Selection
17 // MARK: - RawImagePatchArgs
18
19 typedef struct
20 {
21     void (*progressFunc)(void *key, float progress); // Optional
22     void *progressFuncKey; // Optional
23     const char *inputFile; // Optional. If missing, `*full replacement*`.
24     const char *outputFile;
25     const char *patchFile;
26     bool isNotCryptexCache;
27     uint32_t threadCount;
28     uint32_t verboseLevel;
29 } RawImagePatchArgs;
30
```

Line: 31 Col: 1

Credit to DhinakG



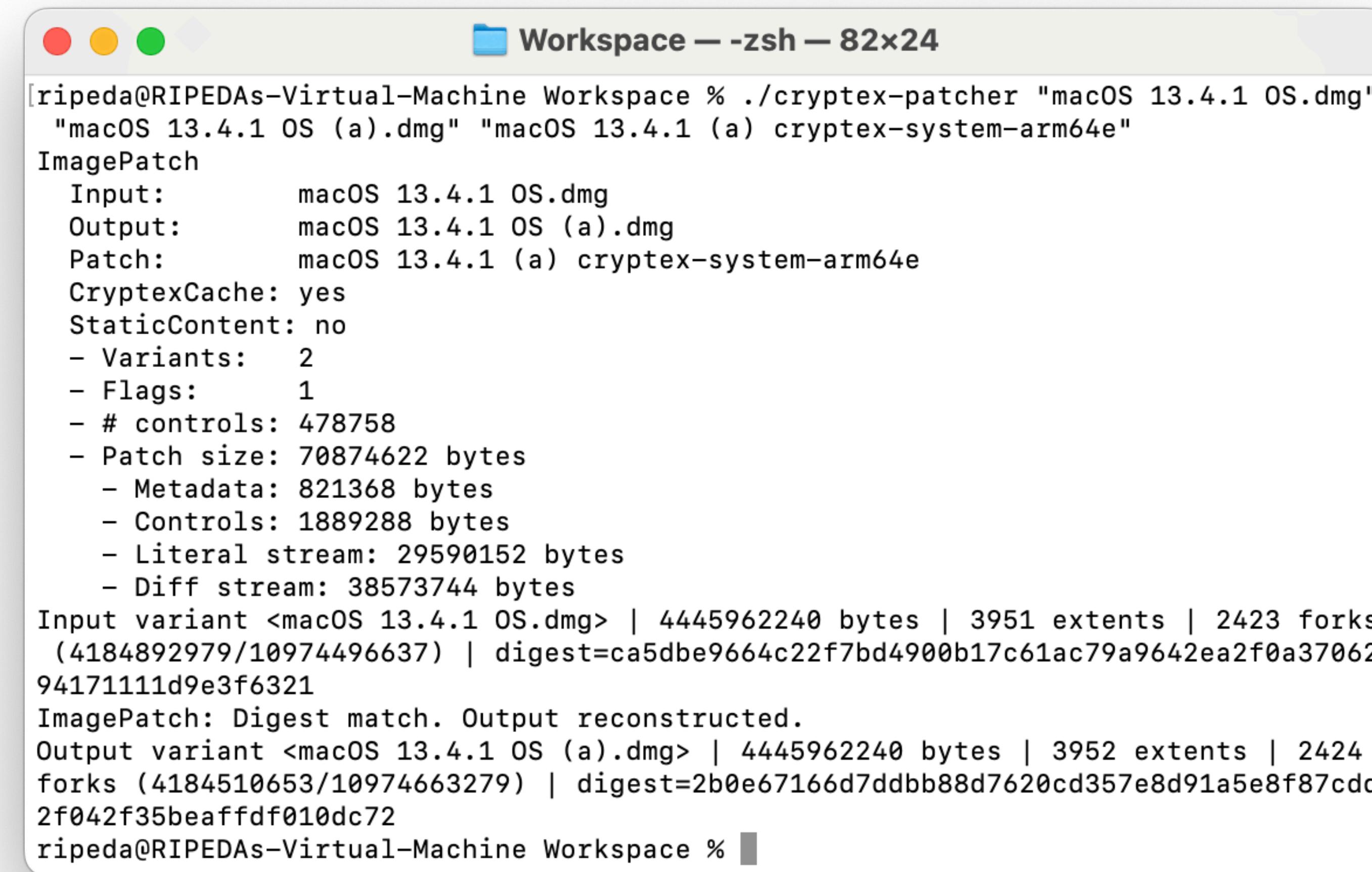
```
extract.c
C extract.c > RawImagePatchArgs
32 // MARK: - RawImagePatch
33
34 int RawImagePatch(RawImagePatchArgs *args) {
35
36     void *lib = dlopen("/usr/lib/libParallelCompression.dylib", RTLD_LAZY);
37     if (!lib) {
38         fprintf(stderr, "Failed to load the library\n");
39         return 1;
40     }
41
42     int (*RawImagePatch)(RawImagePatchArgs*) = dlsym(lib, "RawImagePatch");
43     if (!RawImagePatch) {
44         fprintf(stderr, "Failed to get the function\n");
45         return 1;
46     }
47
48     int result = RawImagePatch(args);
49     if (result != 0) {
50         fprintf(stderr, "The function returned %d\n", result);
51         return 1;
52     }
53
54     dlclose(lib);
55
56     return 0;
57 }
```

Line: 28 Col: 27

Credit to ASentientBot

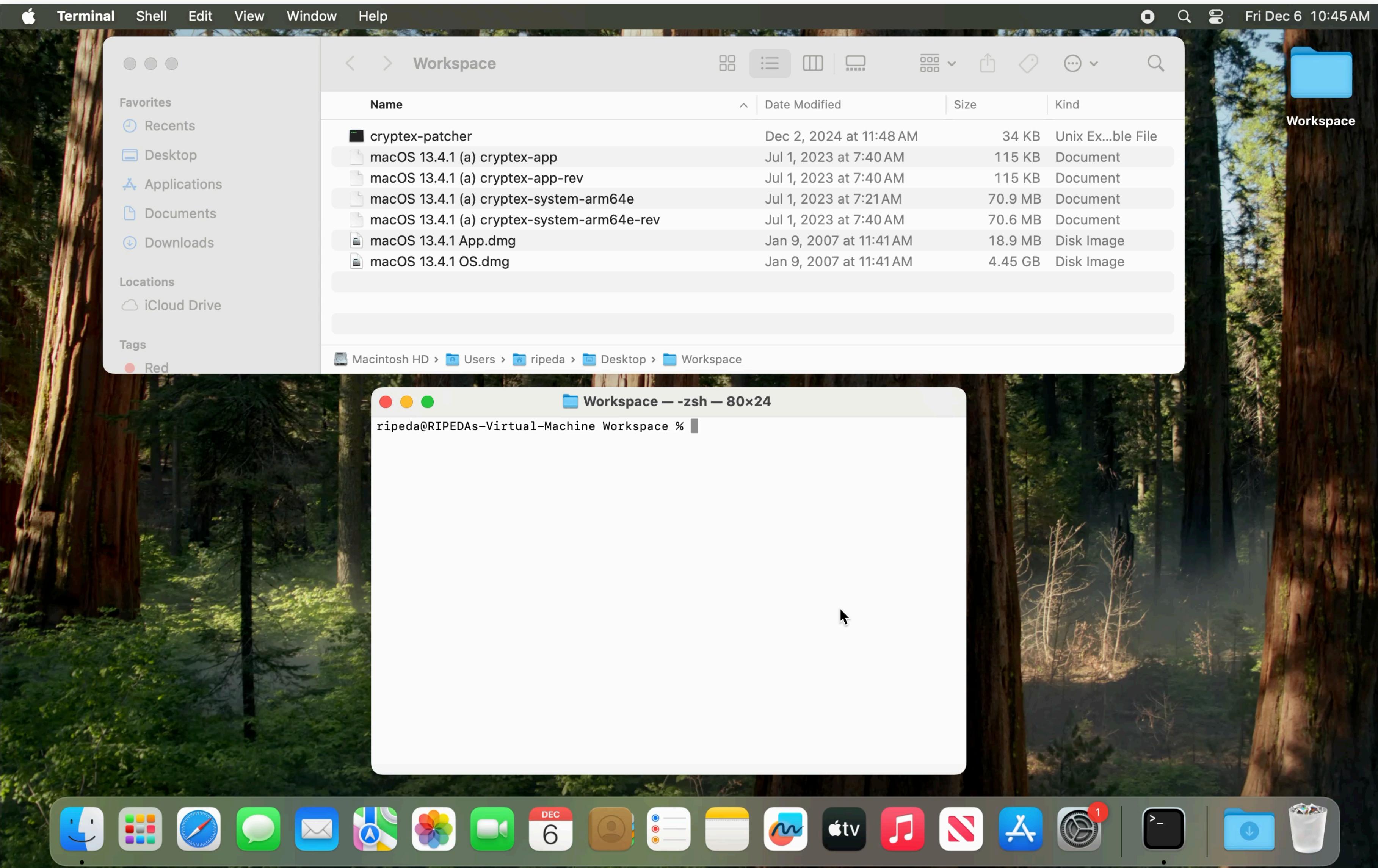
# How do RSRs/Cryptexes work?

**./cryptex-patcher <cryptex>.dmg <output>.dmg <rsr>**

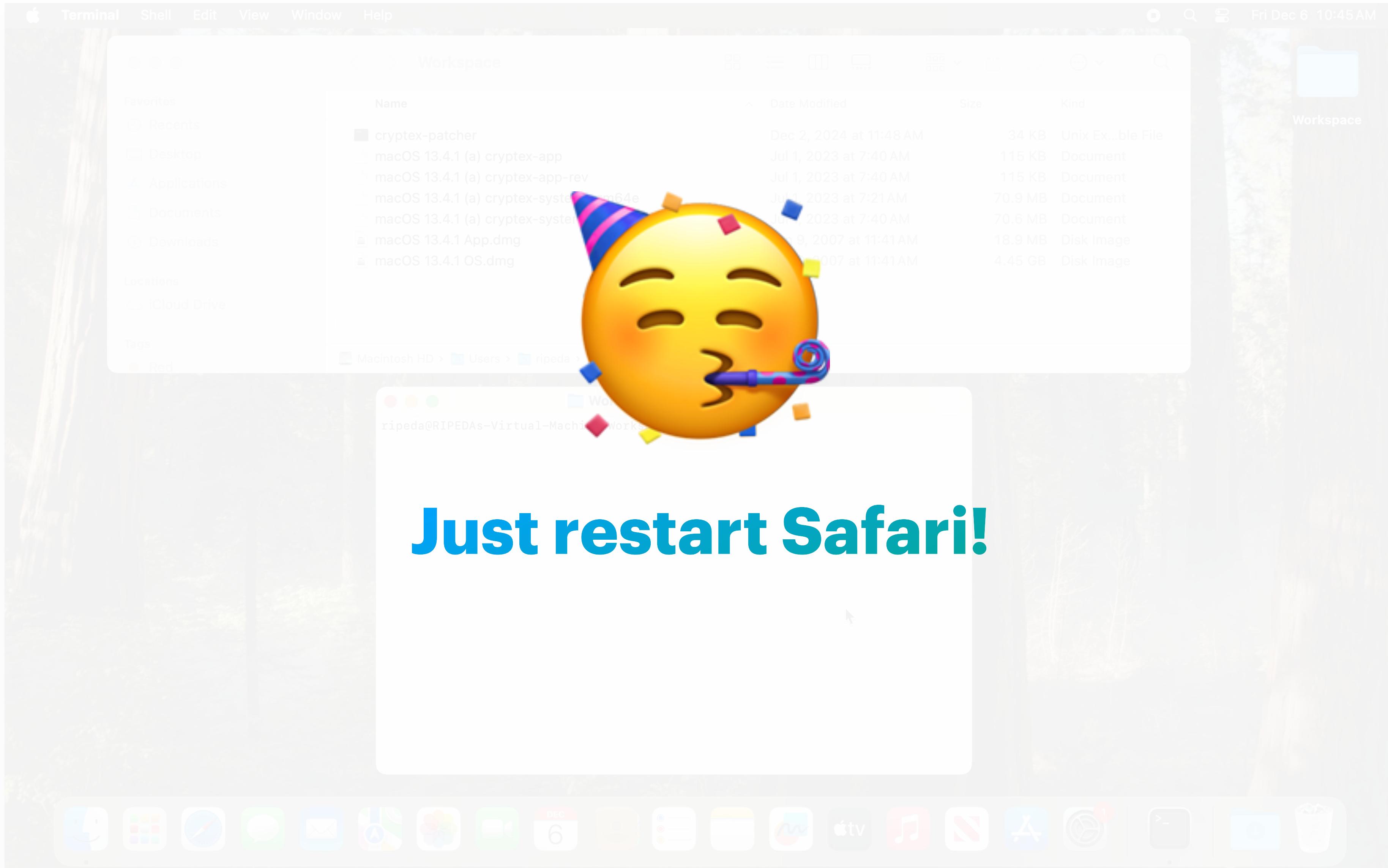


```
[ripeda@RIPEDAs-Virtual-Machine Workspace % ./cryptex-patcher "macOS 13.4.1 OS.dmg"
"macOS 13.4.1 OS (a).dmg" "macOS 13.4.1 (a) cryptex-system-arm64e"
ImagePatch
  Input:      macOS 13.4.1 OS.dmg
  Output:     macOS 13.4.1 OS (a).dmg
  Patch:      macOS 13.4.1 (a) cryptex-system-arm64e
  CryptexCache: yes
  StaticContent: no
  - Variants:  2
  - Flags:    1
  - # controls: 478758
  - Patch size: 70874622 bytes
    - Metadata: 821368 bytes
    - Controls: 1889288 bytes
    - Literal stream: 29590152 bytes
    - Diff stream: 38573744 bytes
Input variant <macOS 13.4.1 OS.dmg> | 4445962240 bytes | 3951 extents | 2423 forks
(4184892979/10974496637) | digest=ca5dbe9664c22f7bd4900b17c61ac79a9642ea2f0a37062
94171111d9e3f6321
ImagePatch: Digest match. Output reconstructed.
Output variant <macOS 13.4.1 OS (a).dmg> | 4445962240 bytes | 3952 extents | 2424
forks (4184510653/10974663279) | digest=2b0e67166d7ddb88d7620cd357e8d91a5e8f87cdd
2f042f35beaffdf010dc72
ripeda@RIPEDAs-Virtual-Machine Workspace %
```

# How do RSRs/Cryptexes work?



# How do RSRs/Cryptexes work?



# How do RSRs/Cryptexes work?

The screenshot shows a file comparison interface comparing two SystemCryptex volumes: RomeF22F82.arm64eSystemCryptex and RomeFRapid22F770820b.arm64eSystemCryptex.

**Top Panel:** Shows the main menu bar with options like New Comparison, Start/Stop, Layout, Comparison, Preferences, Changes, Edits, Merging, Copy Selected, Select Rows, Selected Items, Hide/Reveal, Swap Panes, Folders, and Bookmarks. The title bar indicates the comparison is between RomeF22F82.arm64eSystemCryptex and RomeFRapid22F770820b.arm64eSystemCryptex.

**Left Panel (First Volume):** Displays the directory structure of the first volume. It includes the System and Library folders, with CoreServices being expanded. Inside CoreServices, RestoreVersion.plist and SystemVersion.plist are listed. The dyld folder contains several shared cache files (aot\_shared\_cache.0, aot\_shared\_cache.1, etc.) and dyld\_shared\_cache\_arm64e files.

**Right Panel (Second Volume):** Displays the directory structure of the second volume, mirroring the first but with changes. The System, Library, and CoreServices folders are present. The dyld folder also contains similar shared cache files.

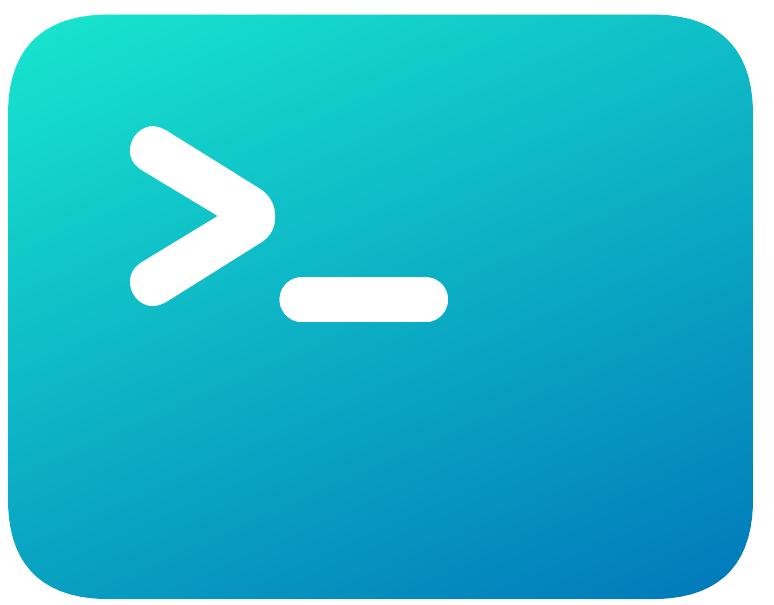
**Bottom Panels:** Two code editors show the contents of SystemVersion.plist. The left editor is for RomeF22F82 and the right for RomeFRapid. The code is XML-based and defines various system properties. Differences are highlighted with color-coded annotations (green for additions, red for deletions).

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
3 <plist version="1.0">
4 <dict>
5   <key>BuildID</key>
6   <string>31B2220C-0B5C-11EE-BFD2-A781FB1AB02B</string>
7   <key>ProductBuildVersion</key>
8   <string>22F82</string>
9   <key>ProductCopyright</key>
10  <string>1983-2023 Apple Inc.</string>
11  <key>ProductName</key>
12  <string>macOS</string>
13  <key>ProductUserVisibleVersion</key>
14  <string>13.4.1</string>
15  <key>ProductVersion</key>
16  <string>13.4.1</string>
17  <key>iOSSupportVersion</key>
18  <string>16.5</string>
19 </dict>
20 </plist>
```

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
3 <plist version="1.0">
4 <dict>
5   <key>BuildID</key>
6   <string>E64ED012-17BD-11EE-A3CB-CD47402D2717</string>
7   <key>ProductBuildVersion</key>
8   <string>22F770820b</string>
9   <key>ProductCopyright</key>
10  <string>1983-2023 Apple Inc.</string>
11  <key>ProductName</key>
12  <string>macOS</string>
13  <key>ProductUserVisibleVersion</key>
14  <string>13.4.1</string>
15  <key>ProductVersion</key>
16  <string>13.4.1</string>
17  <key>ProductVersionExtra</key>
18  <string>(a)</string>
19  <key>iOSSupportVersion</key>
20  <string>16.5</string>
```

At the bottom, status information is displayed: Unicode (UTF-8 without BOM) • 0 removals • 1 insertion • 2 changes • Line 6 of 20 • Column 1.

# dyld\_shared\_cache system



**Test bin**

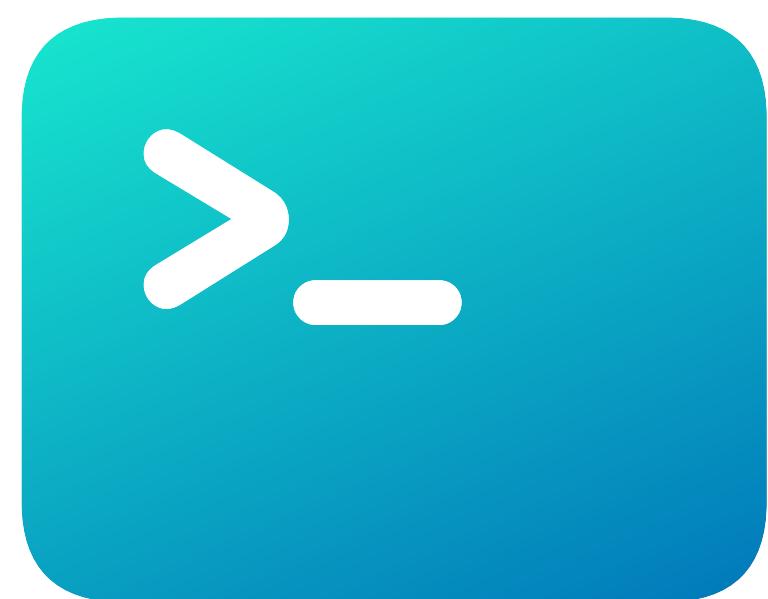


**Demo.framework**

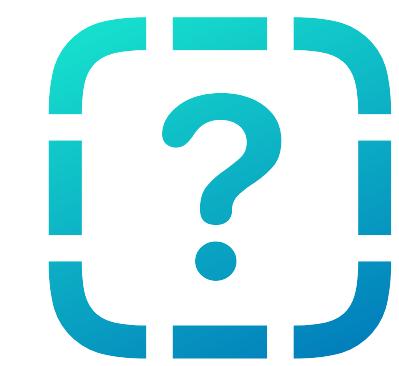


**dyld**

# dyld\_shared\_cache system



**Test bin**



**Demo.framework**



**dyld**

**System volume stand alone ?**

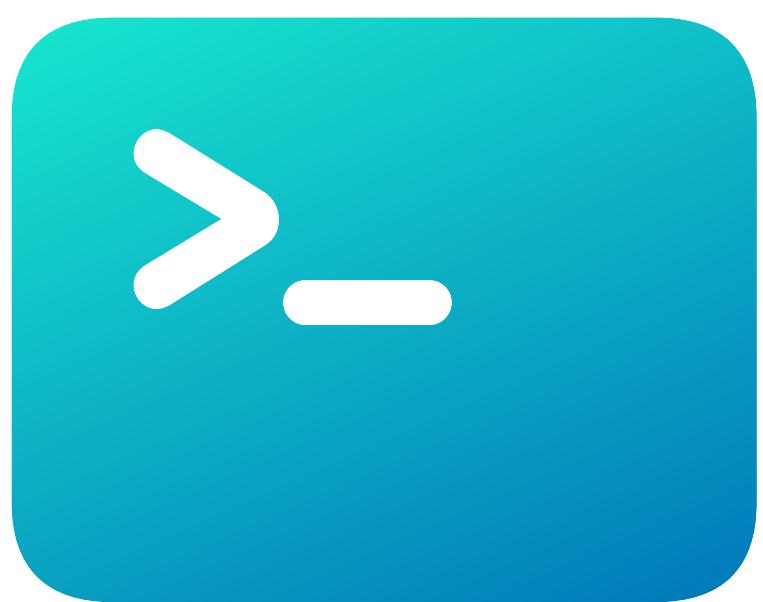
**Cryptex stand alone?**

**System volume dyld\_shared\_cache ?**

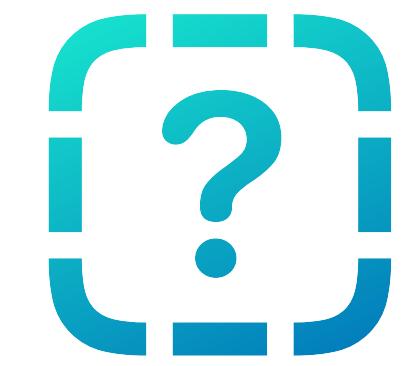
**Cryptex dyld\_shared\_cache ?**

**Relative path?**

# dyld\_shared\_cache system



**Test bin**



**Demo.framework**



**dyld**

**System volume stand alone ?**

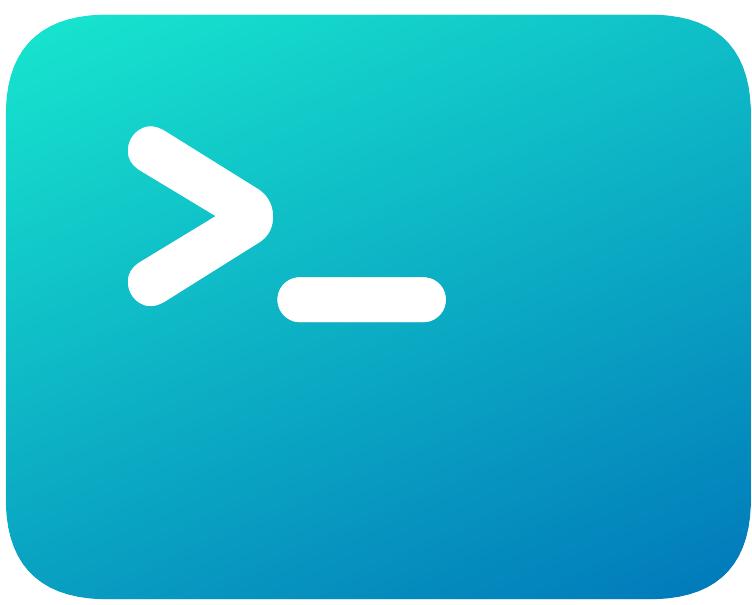
**Cryptex stand alone!**

**System volume dyld\_shared\_cache ?**

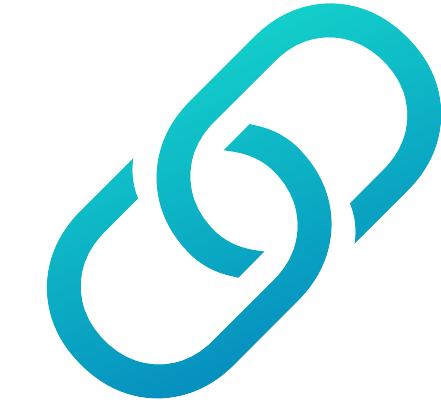
**Cryptex dyld\_shared\_cache ?**

**Relative path?**

# dyld\_shared\_cache system



**Test bin**

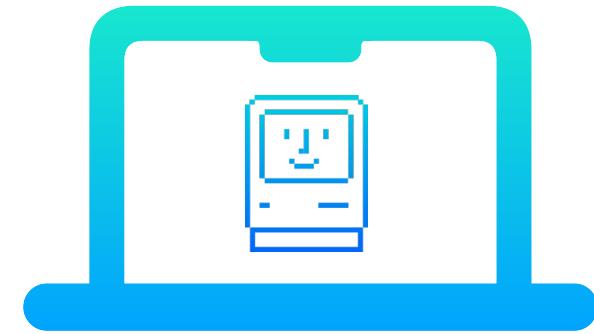


**/System/Cryptex/OS/System/Library  
/Frameworks/Demo.framework**

Lets follow the  
macOS Boot Process

# macOS Boot Process

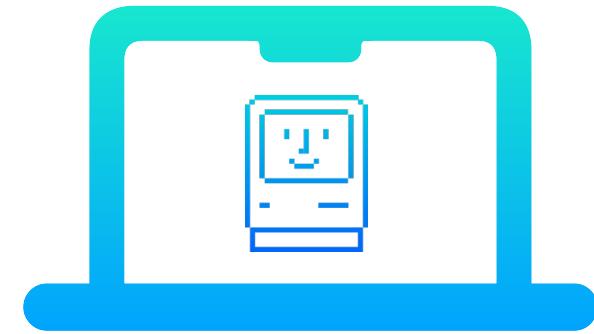
Intel Macs



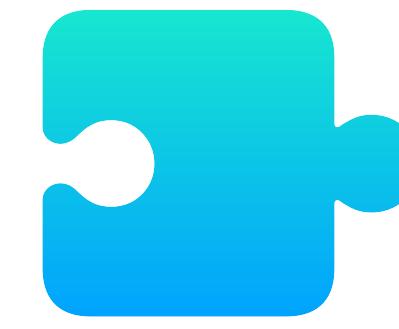
**1. boot.efi and XNU init**

# macOS Boot Process

Intel Macs



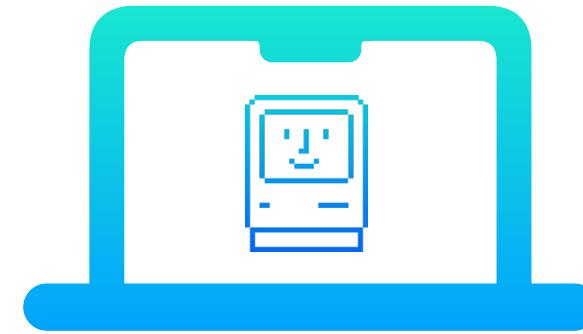
1. boot.efi and XNU init



2. Kernel Extensions load

# macOS Boot Process

Intel Macs



1. boot.efi and XNU init



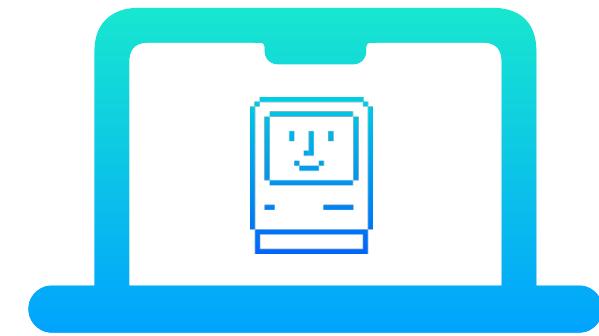
2. Kernel Extensions load



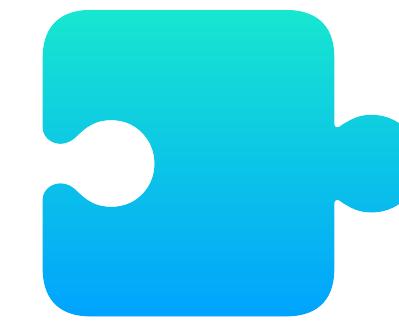
3. apfs.kext starts

# macOS Boot Process

Intel Macs



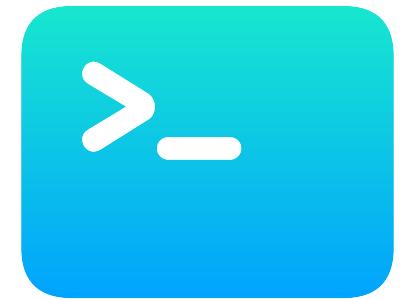
1. boot.efi and XNU init



2. Kernel Extensions load



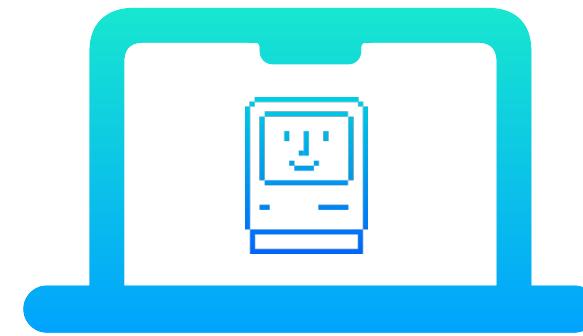
3. apfs.kext starts



4. Userspace starts

# macOS Boot Process

Intel Macs



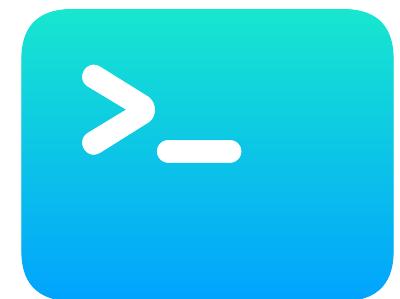
1. boot.efi and XNU init



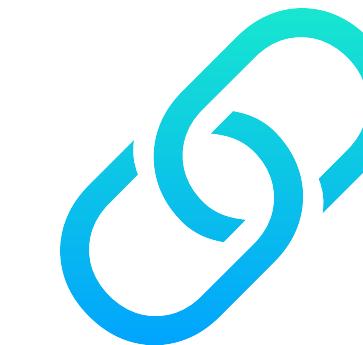
2. Kernel Extensions load



3. apfs.kext starts



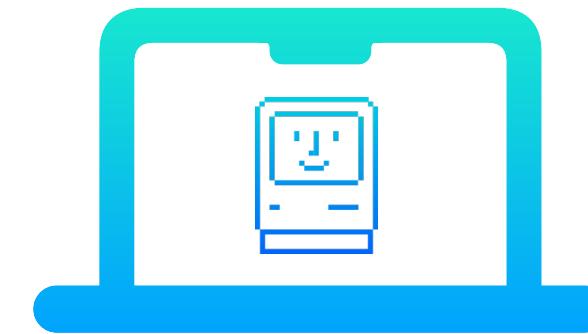
4. Userspace starts



5. /usr/lib/dyld starts

# macOS Boot Process

Intel Macs



1. boot.efi and XNU init



2. Kernel Extensions load



3. apfs.kext starts



4. Userspace starts



5. /usr/lib/dyld starts



6. \_apfs\_graft executed

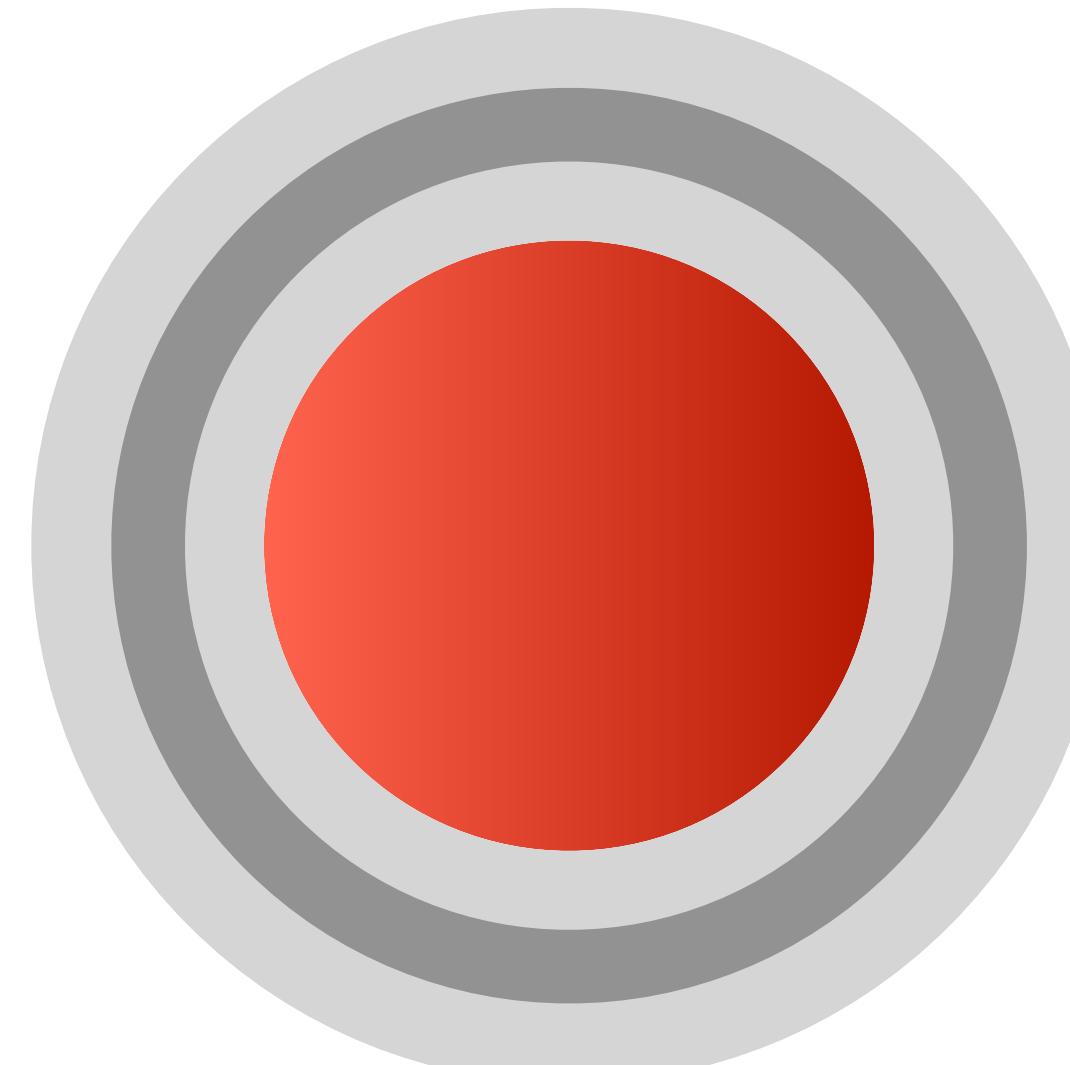
**Where do RSRs fall short?**

# Where do RSRs fall short?

## Kernel Space Updates

# What is kernel space?

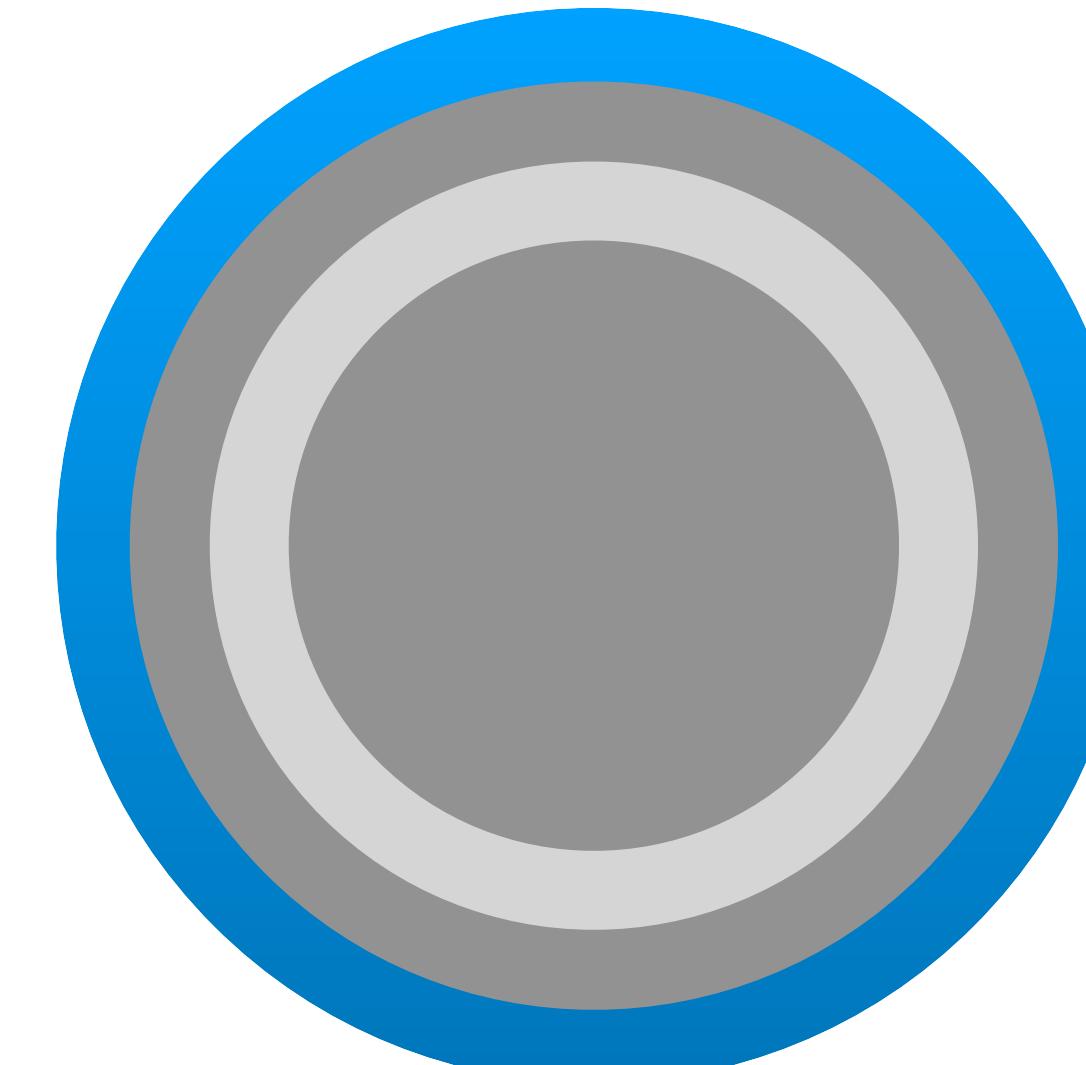
# What is kernel space?



Ring 0

## Kernel Space

- Closer to hardware
- Kernel and Kernel Extensions
- Higher danger when exploited

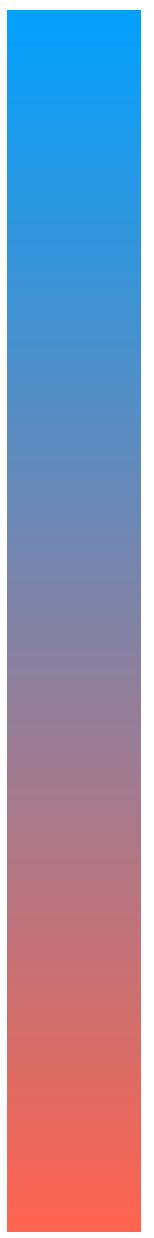


Ring 3

## User Space

- Lower privileges
- Apps, CLIs, etc
- Goes through kernel space

Least Privileged

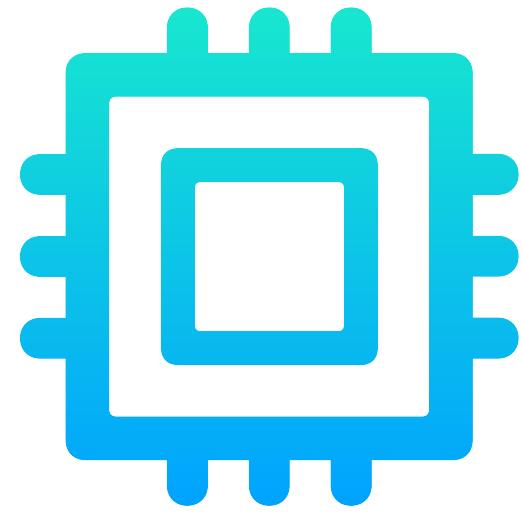


Most Privileged

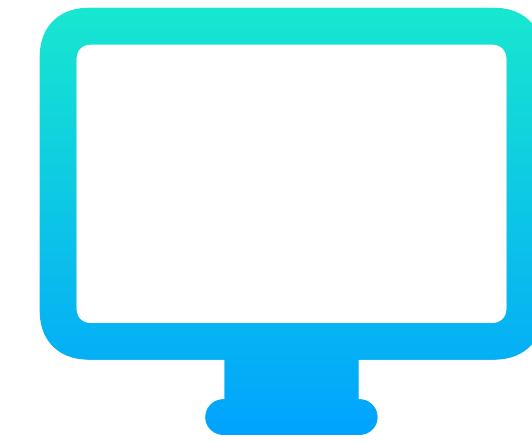
# Apple's 3 stage kernel caching system

# Apple's 3 stage kernel caching system

Introduced with macOS 11, Big Sur, at WWDC 2020



Boot Kernel Collection



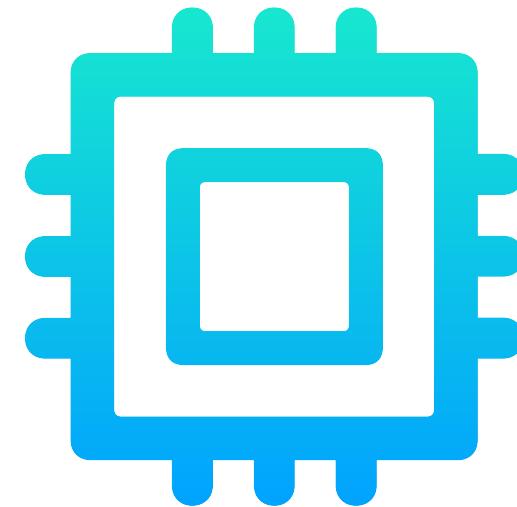
System Kernel Collection



Auxiliary Kernel Collection

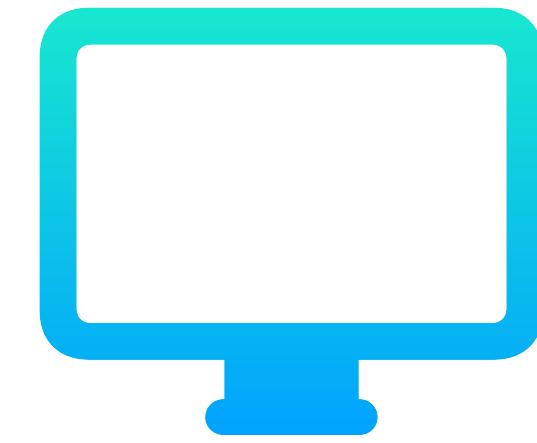
# Apple's 3 stage kernel caching system

Introduced with macOS 11, Big Sur, at WWDC 2020



## Boot Kernel Collection

- Critical to booting
- Kernel, ie. XNU
- Core Drivers (ex. NVMe)
- Preboot volume



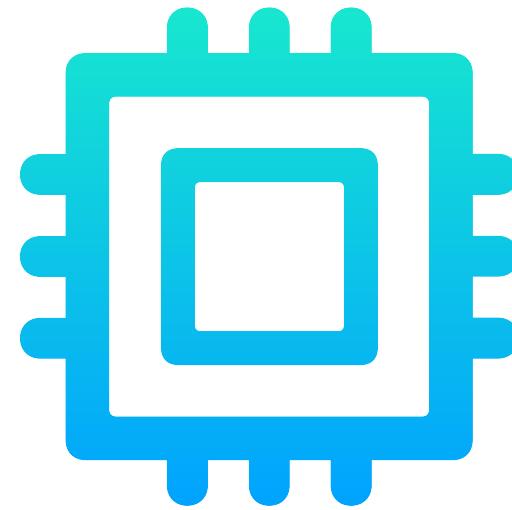
## System Kernel Collection



## Auxiliary Kernel Collection

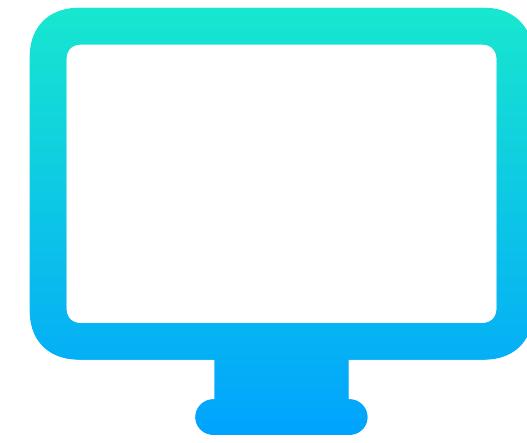
# Apple's 3 stage kernel caching system

Introduced with macOS 11, Big Sur, at WWDC 2020



## Boot Kernel Collection

- Critical to booting
- Kernel, ie. XNU
- Core Drivers (ex. NVMe)
- Preboot volume



## System Kernel Collection

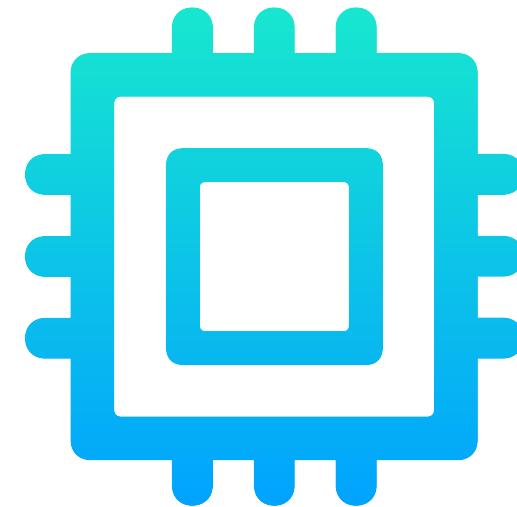
- Pageable
  - ex. Graphics & Audio
  - N/A on Apple Silicon
- System volume



## Auxiliary Kernel Collection

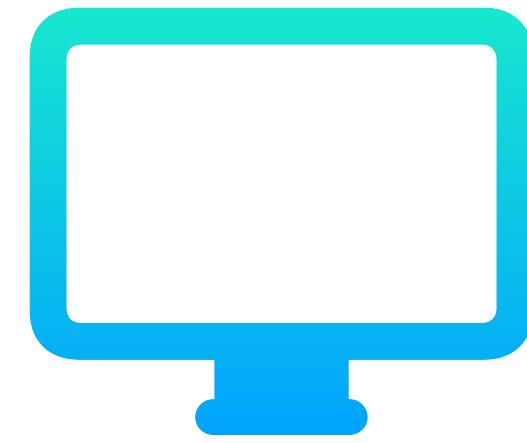
# Apple's 3 stage kernel caching system

Introduced with macOS 11, Big Sur, at WWDC 2020



## Boot Kernel Collection

- Critical to booting
- Kernel, ie. XNU
- Core Drivers (ex. NVMe)
- Preboot volume



## System Kernel Collection

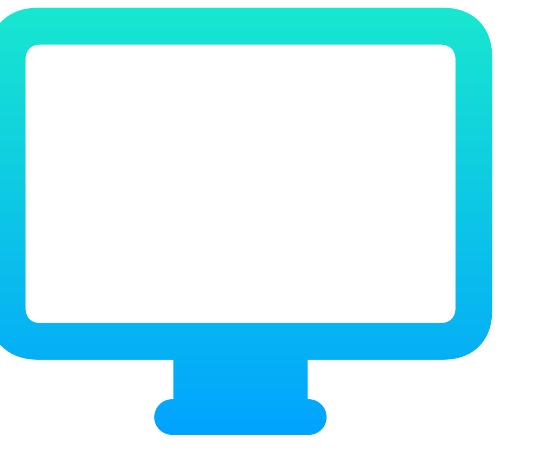
- Pageable
- ex. Graphics & Audio
- N/A on Apple Silicon
- System volume



## Auxiliary Kernel Collection

- 3rd party drivers
- ex. RAID Controllers
- Bridge as DriverKit developed
- Data volume





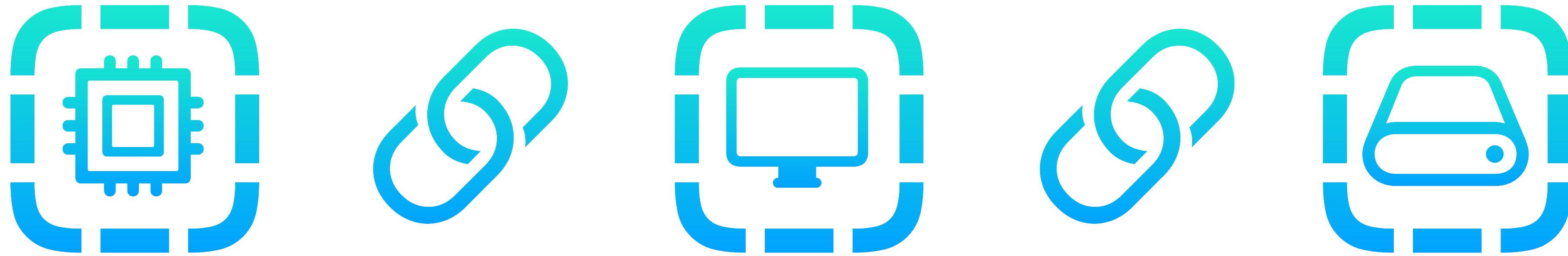
## System Kernel Collection

- System volume



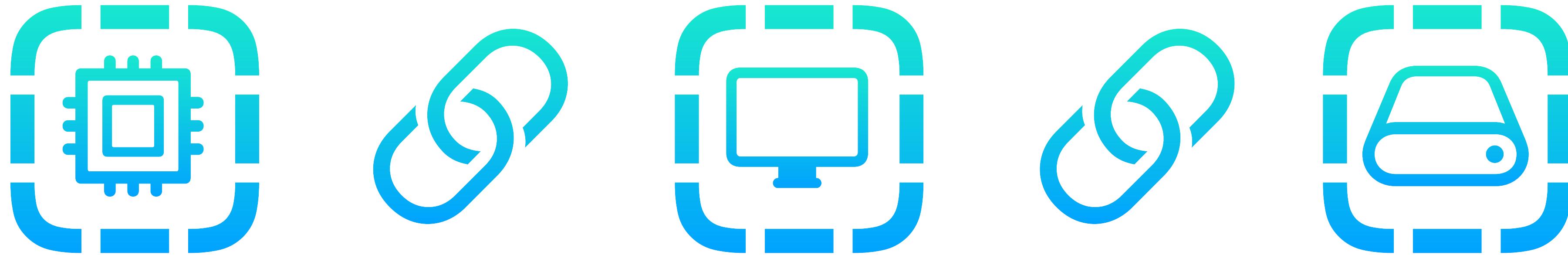
# Kernel Collection UUIDs and APFS Snapshots

# Kernel Collection UUIDs and APFS Snapshots



Linked at build time

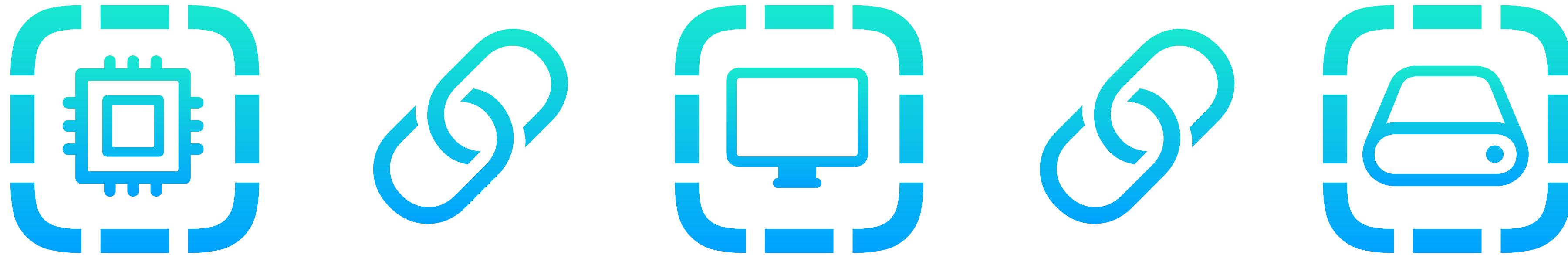
# Kernel Collection UUIDs and APFS Snapshots



Linked at build time

Overlay/hot load kernel extensions from an RSR?

# Kernel Collection UUIDs and APFS Snapshots



Killed with the Kernel Collections Architecture  
in Big Sur...

Overlay/hot load kernel extensions from an RSR?

# Kernel Collection UUIDs and APFS Snapshots



**So why are snapshots an issue?**

Killed with the Kernel Collections Architecture  
in Big Sur...

Overlay/hot load kernel extensions from an RSR?

# Kernel Collection UUIDs and APFS Snapshots



**So why are snapshots an issue?**

**Multiple reboots to apply...**

Killed with the Kernel Collections Architecture  
in Big Sur...

Overlay/hot load kernel extensions from an RSR?

# Why are RSRs no longer used?

# Why are RSRs no longer used?

macOS 13.4.1 (a)...(b)...(c)...

iOS 16.5.1 (a)...(b)...(c)...

# macOS 13.4.1 (& iOS 16.5.1) Timeline



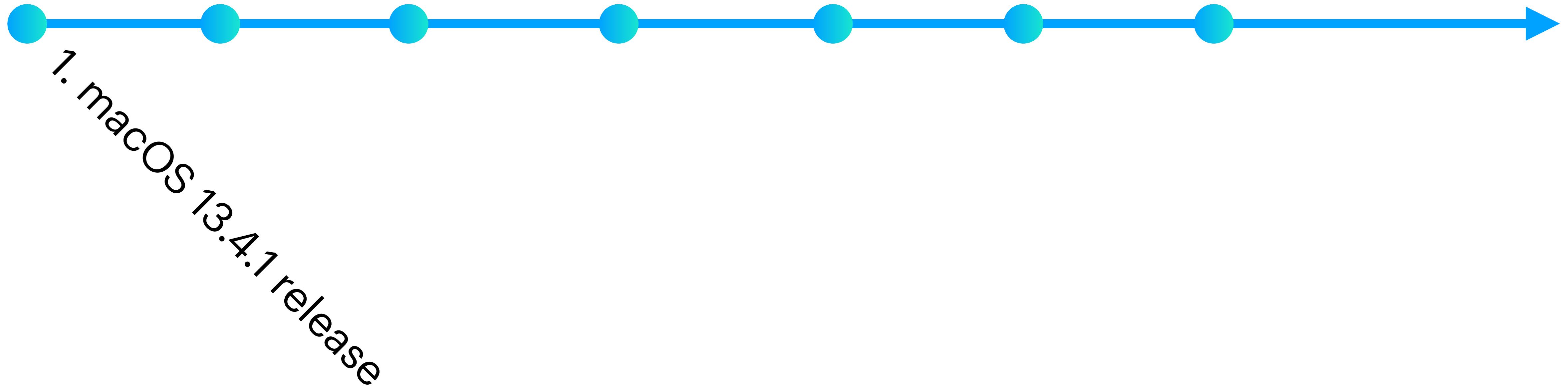
# macOS 13.4.1 (& iOS 16.5.1) Timeline



**macOS 13.4.1 (& iOS 16.5.1) release**

June 21, 2023

# macOS 13.4.1 (& iOS 16.5.1) Timeline



# macOS 13.4.1 (& iOS 16.5.1) Timeline



# macOS 13.4.1 (& iOS 16.5.1) Timeline



# macOS 13.4.1 (& iOS 16.5.1) Timeline



# macOS 13.4.1 (& iOS 16.5.1) Timeline



# macOS 13.4.1 (& iOS 16.5.1) Timeline

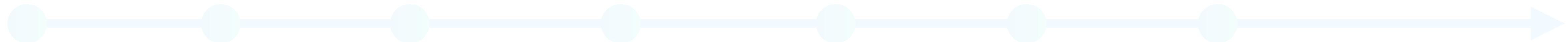


July 10, 2023

# macOS 13.4.1 (& iOS 16.5.1) Timeline



# macOS 13.4.1 (& iOS 16.5.1) Timeline

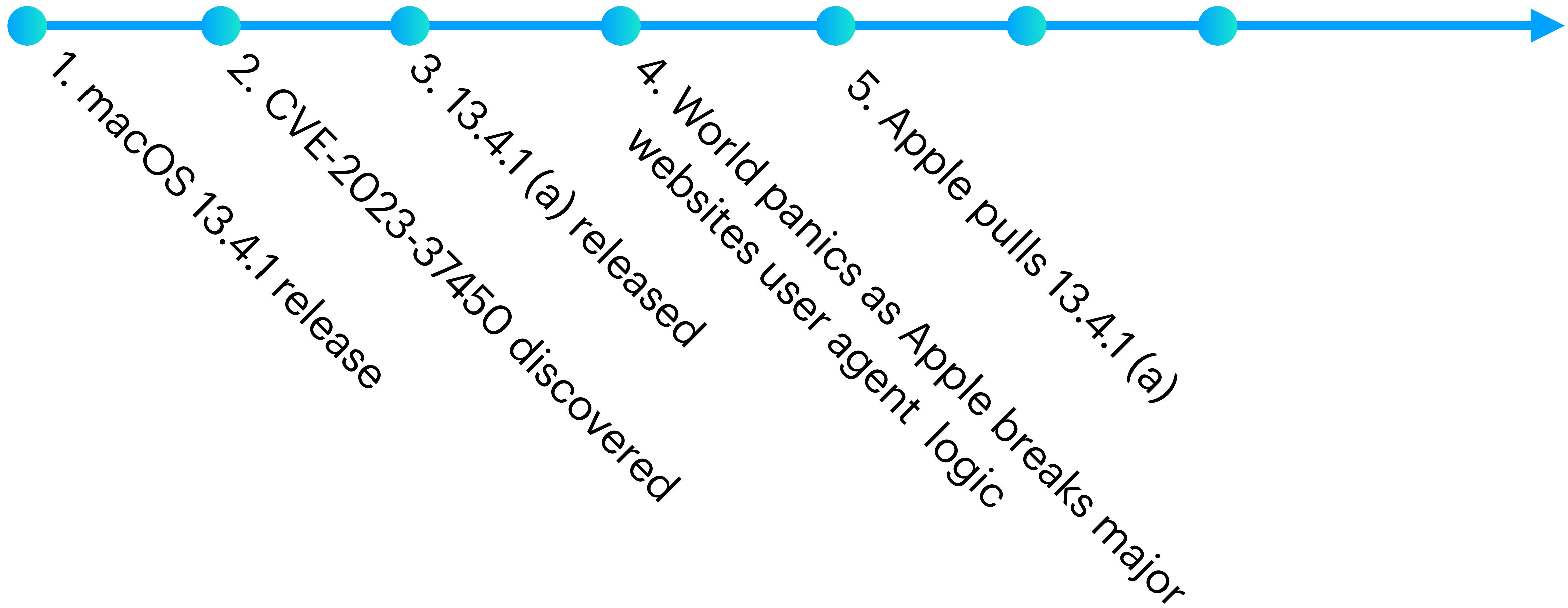


**Apple pulls 13.4.1 (a) and recommends users to uninstall the update if affected**

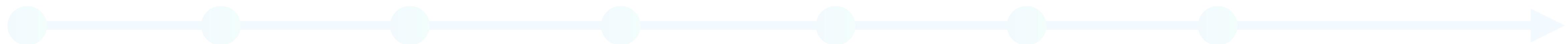
July 10, 2023

1. macOS 13.4.1 release  
2. CVE-2023-37450 discovered  
3. Apple pulls 13.4.1 (a) released  
4. Websites patch as Apple breaks major user agent logic

# macOS 13.4.1 (& iOS 16.5.1) Timeline

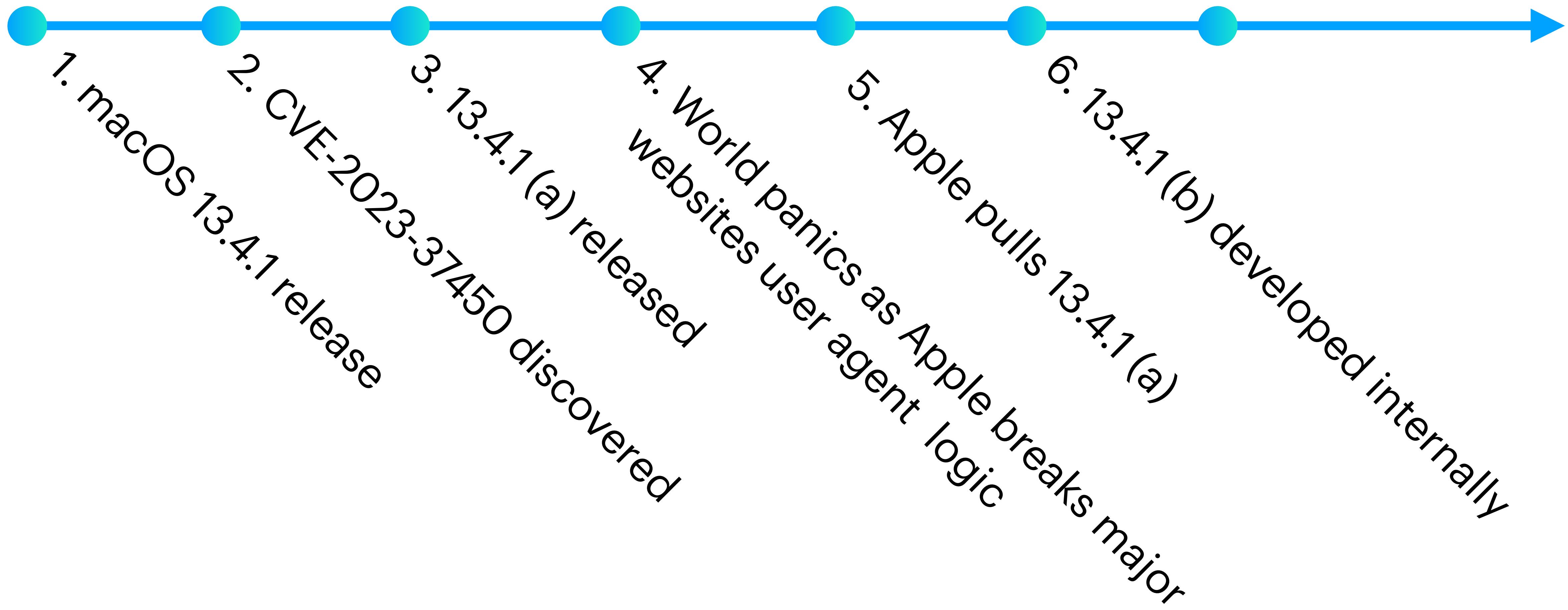


# macOS 13.4.1(&iOS 16.5.1) Timeline



1. macOS 13.4.1 release
  2. CVE-2023-37450 discovered
  3. 13.4.1(a) released
  4. WebKit websites user agent logic
  5. Apple pulls 13.4.1(a)
- 13.4.1 (b) developed internally**
- July 10-11~, 2023

# macOS 13.4.1 (& iOS 16.5.1) Timeline



# macOS 13.4.1(&iOS 16.5.1) Timeline

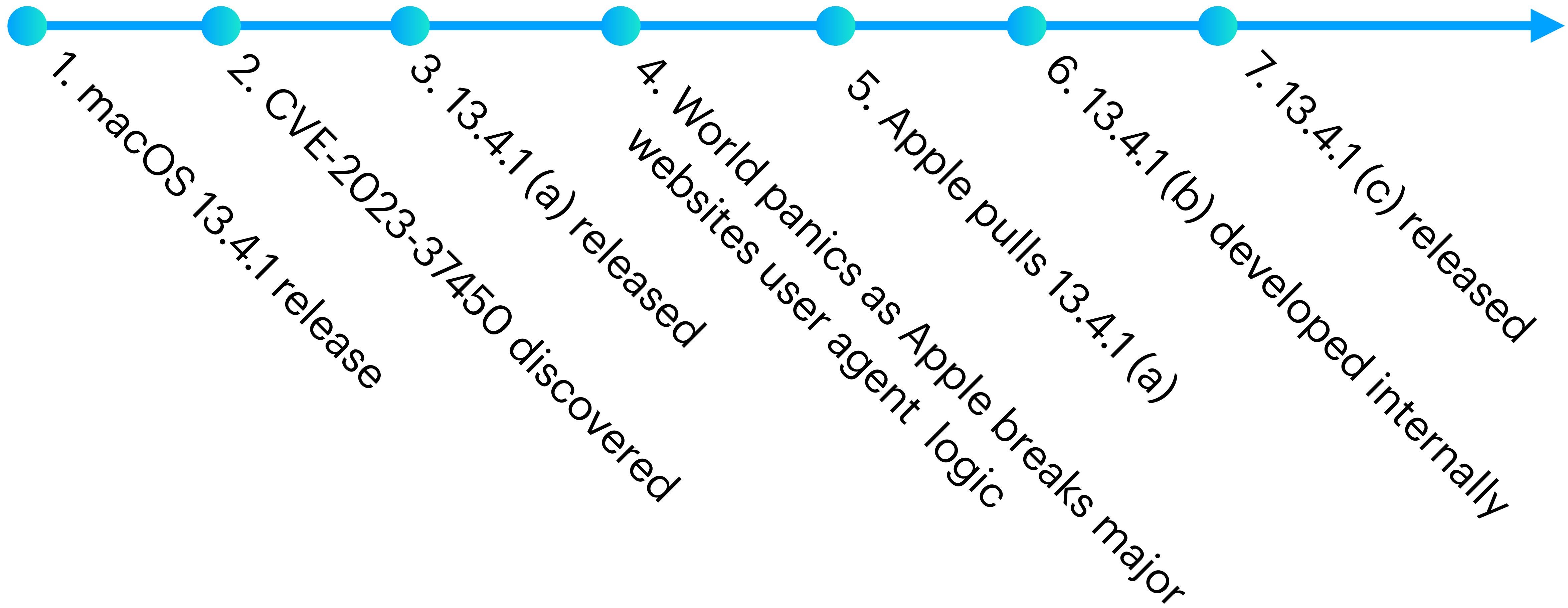


1. macOS 13.4.1 release
2. CVE-2023-37450 discovered
3. 13.4.1(a) released
4. Hand-picked as Apple breaks major logic
5. Apple pulls 13.4.1(a)
6. 13.4.1(b) developed internally

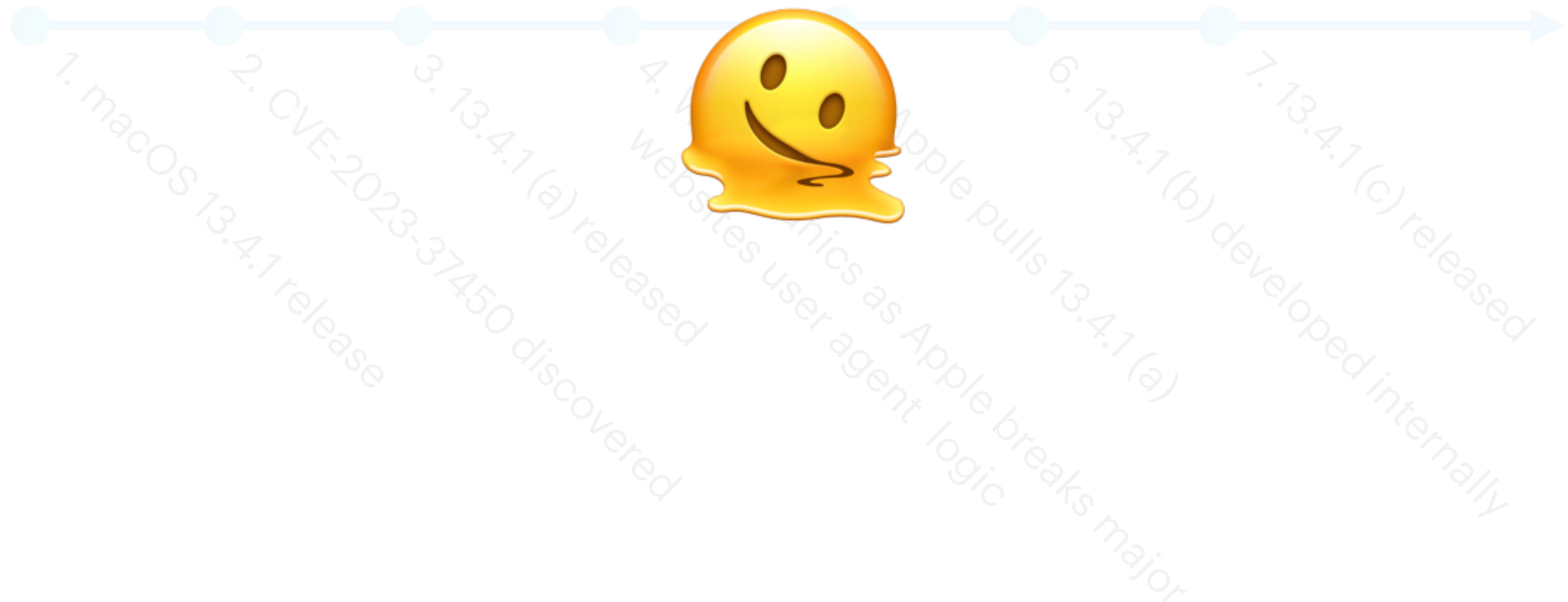
**13.4.1 (c) released** 🎉

July 12, 2023

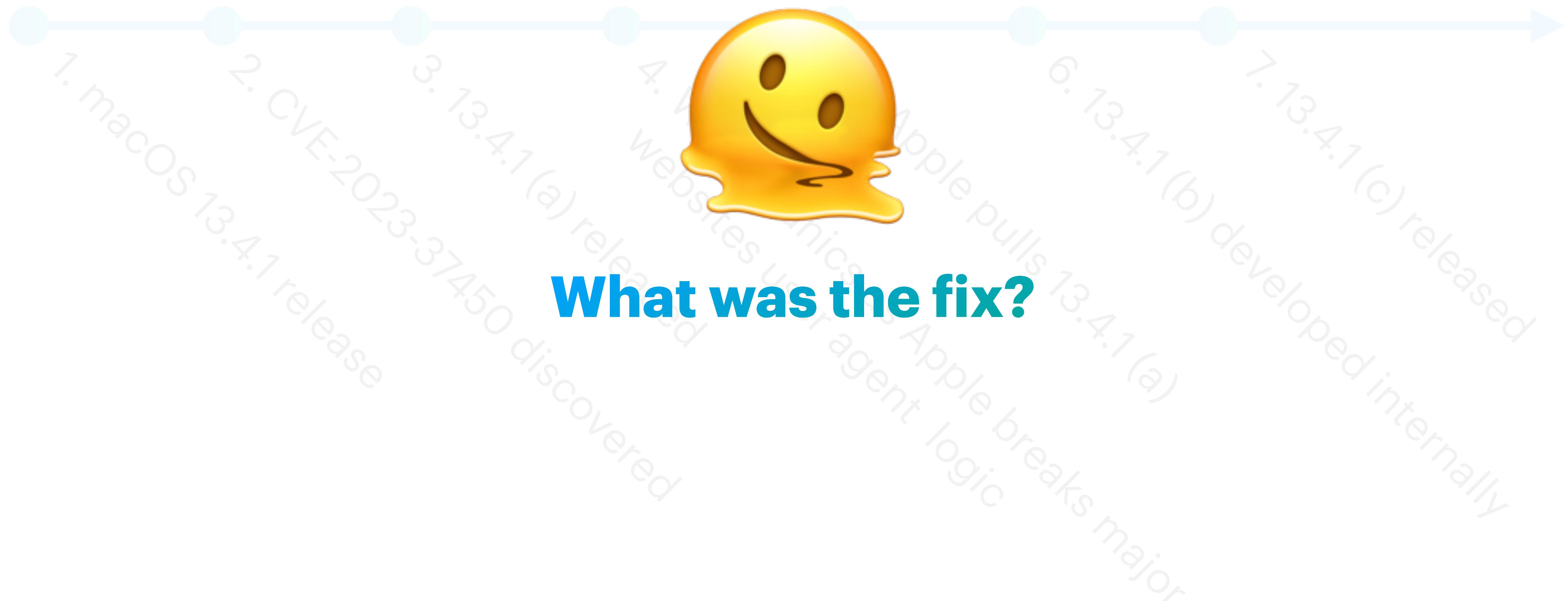
# macOS 13.4.1 (& iOS 16.5.1) Timeline



# macOS 13.4.1(&iOS 16.5.1) Timeline



# macOS 13.4.1(&iOS 16.5.1) Timeline



# macOS 13.4.1(& iOS 16.5.1) Timeline



1. macOS 13.4.1 release
2. CVE-2023-37450 discovered
3. 13.4.1(a) released
4. Websites break
5. Apple pulls 13.4.1(a)
6. 13.4.1(b) developed internally
7. 13.4.1(c) released



**What was the fix?**

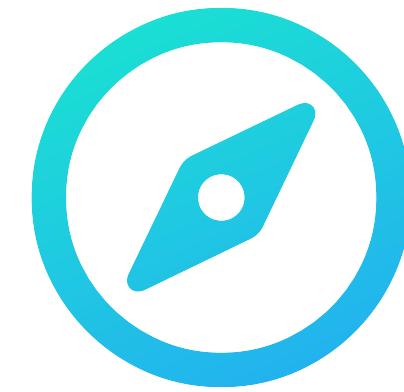
**Removing letters from user-agent versioning...**

**Safari 13.4.1 (a) -> Safari 13.4.1**

**Where are RSRs\* secretly hiding  
today?**

**Where are RSR technologies  
secretly hiding today?**

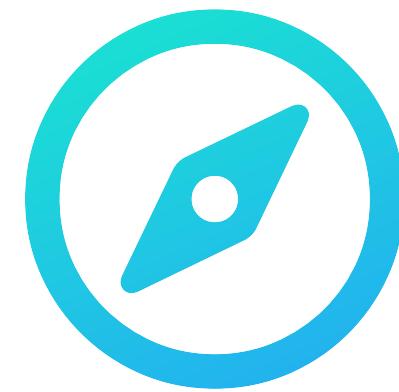
# Where are RSR technologies secretly hiding today?



Safari Updates

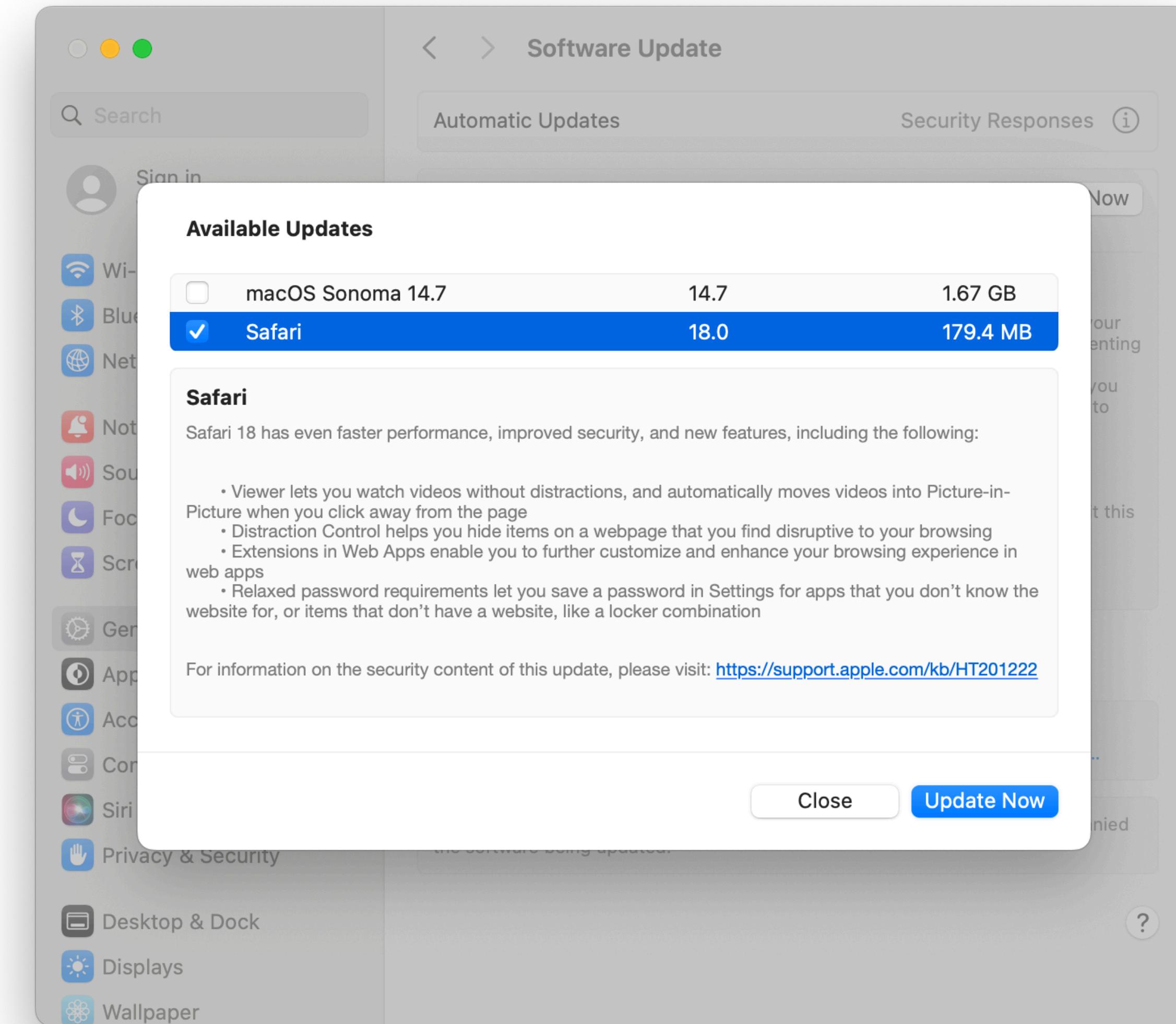


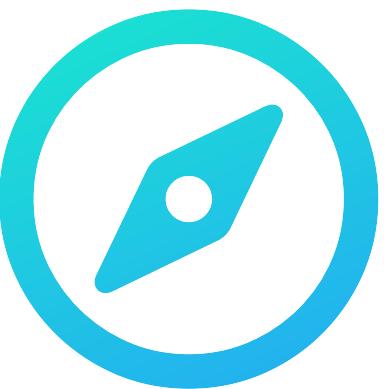
Private Cloud  
Compute



# Safari Updates

## N-1 & N-2 Security Updates





# Safari Updates

<https://swscan.apple.com/content/catalogs/others/index-14-13-12-10.16-10.15-10.14-10.13-10.12-10.11-10.10-10.9-mountainlion-lion-snowleopard-leopard.merged-1.sucatalog>

The screenshot shows the 'Software Update' window in the macOS Settings app. At the top, there are navigation arrows and a title 'Software Update'. Below the title, there are tabs for 'Automatic Updates' and 'Security Responses'. A search bar is located at the top left. On the left side, there's a sidebar with various system settings icons: Sign in, Wi-Fi, Bluetooth, Network, Notifications, Sounds, Focus, Screen Mirroring, General, Apps, Accounts, Control Center, Siri, Privacy & Security, Desktop & Dock, Displays, and Wallpaper. The main content area is titled 'Available Updates' and lists two items: 'macOS Sonoma 14.7' and 'Safari'. The 'Safari' item is selected, indicated by a blue background and a checked checkbox icon. It shows the version as '18.0' and the file size as '179.4 MB'. Below the list, a section titled 'Safari' provides a summary of the update: 'Safari 18 has even faster performance, improved security, and new features, including the following:' followed by a bulleted list of features. At the bottom right of the main window are 'Close' and 'Update Now' buttons.

Available Updates

<input type="checkbox"/>	macOS Sonoma 14.7	14.7	1.67 GB
<input checked="" type="checkbox"/>	Safari	18.0	179.4 MB

**Safari**

Safari 18 has even faster performance, improved security, and new features, including the following:

- Viewer lets you watch videos without distractions, and automatically moves videos into Picture-in-Picture when you click away from the page
- Distraction Control helps you hide items on a webpage that you find disruptive to your browsing
- Extensions in Web Apps enable you to further customize and enhance your browsing experience in web apps
- Relaxed password requirements let you save a password in Settings for apps that you don't know the website for, or items that don't have a website, like a locker combination

For information on the security content of this update, please visit: <https://support.apple.com/kb/HT201222>

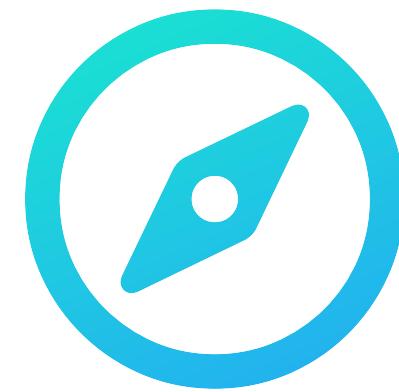
**Close** **Update Now**

index-14-13-12-10.16-10.15-10.14-10.13-10.12-10.11-10.10-10.9-mountainlion-lion-snowleopard-leopard.merged-1.sucatalog

index-14-13-12-10.16-10.15-10.14-10.13-10.12-10.11-10.10-10.9-mountainlion-lion-snowleopard-leopard.merged-1.sucatalog > K 052-58397

Find Text + Aa Contains Done

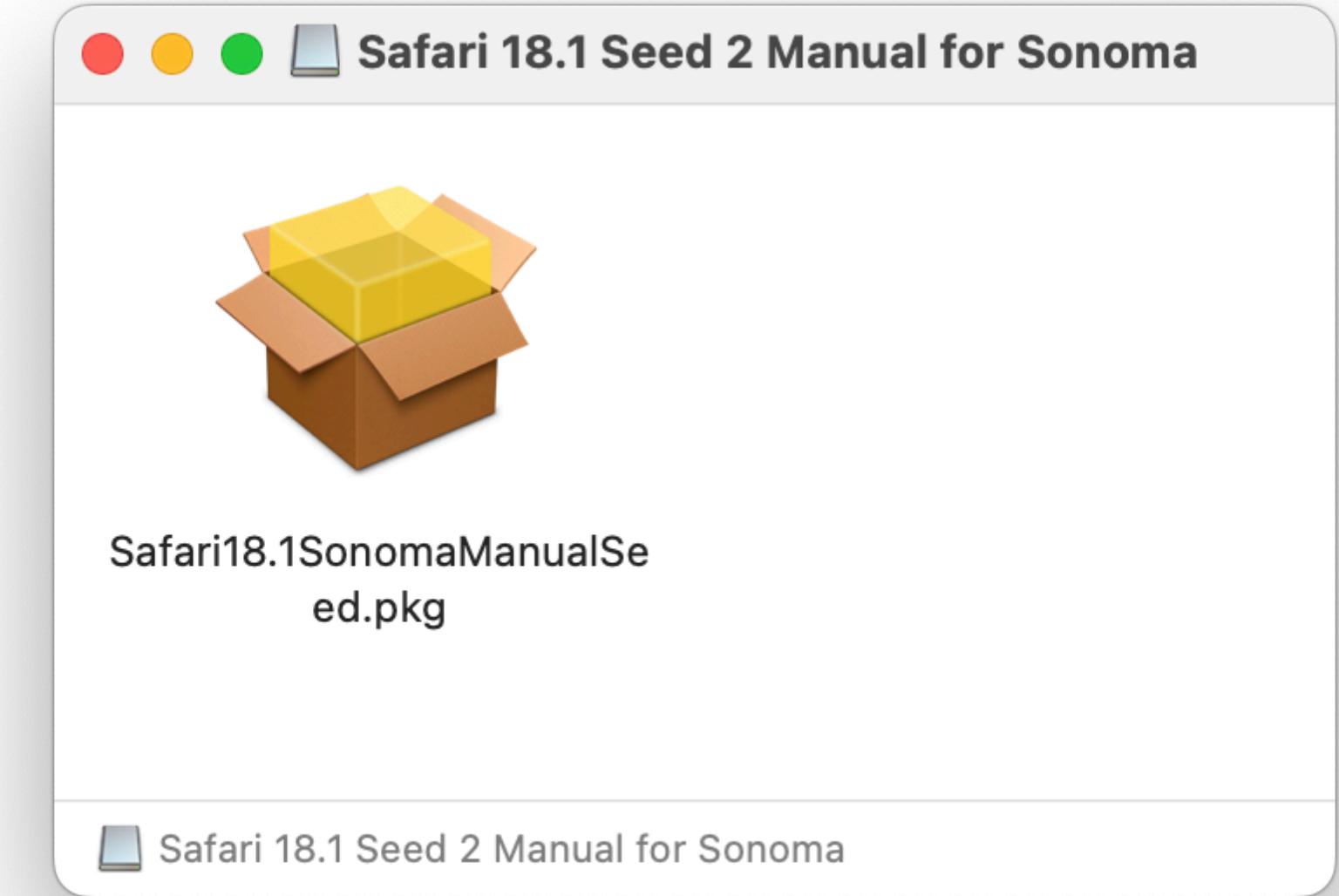
```
4     <dict>
11         <key>Products</key>
96502             <key>052-58397</key>
96503                 <dict>
96504                     <key>ServerMetadataURL</key>
96505                         <string>https://swcdn.apple
96506                             .com/content/downloads/08/48/052-58397-A_R6VSA403XU/gbtv65tb8icy6f3gbg2n6iz
96507                             xj18jtfqmp7/Safari18.0SonomaAuto.smd</string>
96508                     <key>State</key>
96509                         <string>ramped</string>
96510                     <key>Packages</key>
96511                         <array>
96512                             <dict>
96513                                 <key>Digest</key>
96514                                     <string>d51ebddffc9160800e00135bea4c15dc25b32f49</string>
96515                                 <key>Size</key>
96516                                     <integer>179408093</integer>
96517                                 <key>MetadataURL</key>
96518                                     <string>https://swdist.apple
96519                                         .com/content/downloads/08/48/052-58397-A_R6VSA403XU/gbtv65tb8icy6f3
96520                                         gbg2n6izxj18jtfqmp7/Safari18.0SonomaAuto.pkg</string>
96521                         <key>URL</key>
96522                             <string>https://swcdn.apple
96523                                 .com/content/downloads/08/48/052-58397-A_R6VSA403XU/gbtv65tb8icy6f3
96524                                 gbg2n6izxj18jtfqmp7/Safari18.0SonomaAuto.pkg</string>
96525                         </dict>
96526                     </array>
96527                 <key>ExtendedMetaInfo</key>
96528                     <dict>
96529                         <key>ProductType</key>
96530                             <string>Safari</string>
96531                         </dict>
96532                     <key>PostDate</key>
96533                         <date>2024-09-16T17:48:29Z</date>
96534                     <key>Distributions</key>
96535                     <dict>
96536                         <key>hi</key>
96537                             <string>https://swdist.apple
96538                                 .com/content/downloads/08/48/052-58397-A_R6VSA403XU/gbtv65tb8icy6f3gbg2
96539                                 n6izxj18jtfqmp7/052-58397.hi.dist</string>
```

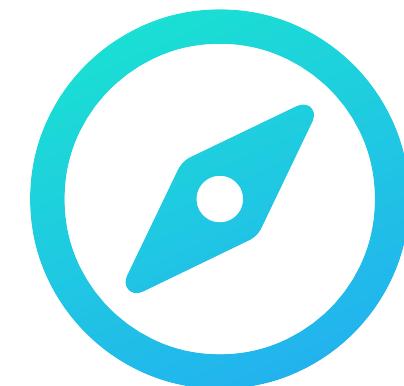


# Safari Updates

<https://developer.apple.com/download/all/?q=Safari>

The screenshot shows a Mac OS X desktop environment. A browser window is open at <https://developer.apple.com/download/all/?q=Safari>. The title bar of the browser shows the URL and the developer.apple.com logo. The main content area is titled "Downloads". At the top, there are navigation links: "Operating Systems", "Applications", "Profiles and Logs", and "More". Below this, the user "Mykola Grymalyuk" is logged in, with a "Sign Out" link. The main heading "More Downloads" is displayed. A search bar below it contains the placeholder "Filter by keywords". In the bottom left corner, there is a small icon of the classic blue and red compass rose from the original Mac OS X interface. The main content area displays a card for the "Safari 18.1 for macOS Sonoma and Safari 18.1 for macOS Ventura beta 2" update, released on September 26, 2024. A "View Details" button is visible at the bottom of this card.





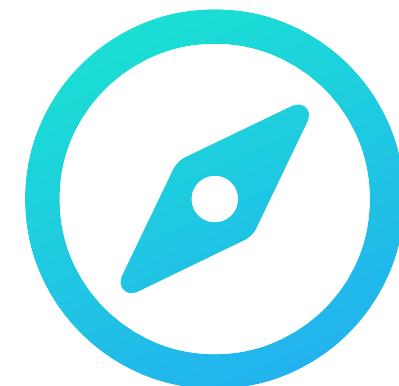
# Safari Updates

Back/Forward      Safari18.1SonomaManualSeed.pkg      Path Action Get Info Quick Look Installer      Search      Exports Review

Package Info      All Files      All Scripts      Receipts

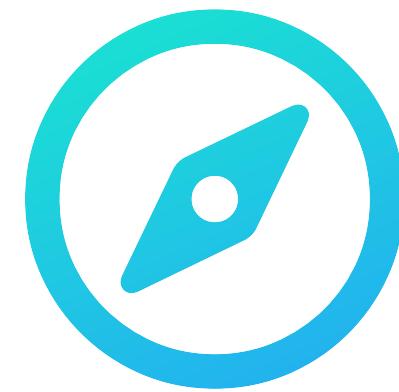
Name	Date Modified	Size	Kind
Library	--	227.2 MB	Folder
Apple	--	227.2 MB	Folder
Safari	--	227.2 MB	Folder
Cryptex	12/31/69	227.2 MB	Folder
062-42150-009.dmg	12/31/69	226.5 MB	Disk Image
Restore	12/31/69	737 KB	Folder
BuildManifest.plist	12/31/69	554 KB	XML property list
Firmware	12/31/69	161 KB	Folder
062-42150-009.dmg.root_hash	12/31/69	229 bytes	Document
062-42150-009.dmg.trustcache	12/31/69	43 bytes	Document
Manifests	12/31/69	161 KB	Folder
> Cryptex1Boot	12/31/69	56 KB	Folder
> restore	12/31/69	105 KB	Folder
> cryptex1	12/31/69	105 KB	Folder
Restore.plist	12/31/69	22 KB	XML property list

All Files      1 item, 227.2 MB installed



# Safari Updates

Name	Date Modified	Size	Kind
>  .fsevents.d	Today at 12:32 PM	--	Folder
✓  Library	Today at 12:32 PM	--	Folder
>  Application Support	Sep 24, 2024 at 10:45 AM	--	Folder
>  Google	Sep 24, 2024 at 10:45 AM	--	Folder
>  Preferences	Today at 12:32 PM	--	Folder
✓  System	Today at 12:32 PM	--	Folder
✓  Applications	Today at 12:32 PM	--	Folder
Safari.app	Sep 24, 2024 at 10:45 AM	13.6 MB	Application
>  Library	Sep 24, 2024 at 10:45 AM	--	Folder
✓  usr	Today at 12:32 PM	--	Folder
✓  bin	Sep 24, 2024 at 10:45 AM	--	Folder
safaridriver	Sep 24, 2024 at 10:45 AM	102 KB	Unix Executable File
>  libexec	Sep 24, 2024 at 10:45 AM	--	Folder
>  share	Sep 24, 2024 at 10:45 AM	--	Folder



# Safari Updates

Back/Forward      Safari18.1SonomaManualSeed.pkg      Path      Action      Get Info      Quick Look      Installer      Search      Export      Review

Package Info      All Files      graftCryptex.sh      Receipts

Safari18.1Sonoma.pkg

- preinstall
- preinstall\_actions
  - cleanupCryptex.sh
  - cleanUpgrade
- postinstall
- postinstall\_actions
  - clearxattr
  - graftCryptex.sh
  - loadSafariBookmarksSyncAgent
  - notifySiri
  - ReloadXPCCache
  - setACL
  - UnloadSafariHistory
  - UnloadSafariNotificationAgent
- safariFiles
- systemFiles
- Tools
  - deleteomatic

graftCryptex.sh

```
#!/bin/sh
targetVolume="$3"
cryptegraft="$targetVolume/System/Library/PrivateFrameworks/MobileSoftwareUpdate.framework/Support/cryptegraft"
cryptex_dir="$targetVolume/Library/Apple/Safari/Cryptex"
if [ ! -f "$cryptegraft" ] || [ ! -x "$cryptegraft" ]; then
    echo "Unable to find and execute '$cryptegraft'"
    exit 1
fi
if [ ! -d "$cryptex_dir" ]; then
    echo "The Cryptex directory does not exist at '$cryptex_dir'"
    exit 2
fi
echo "Installing the Cryptex"
"$cryptegraft" --downlevel --updateBundle "$cryptex_dir/Restore" --targetVolume "$targetVolume"
install_status=$?
# rdar://110673814
if [ -f "/System/Cryptexes/App/Library/Google/Chrome/NativeMessagingHosts/com.apple.passwordmanager.json" ]; then
    mkdir -p "/Library/Google/Chrome/NativeMessagingHosts"
    ln -sf /System/Cryptexes/App/Library/Google/Chrome/NativeMessagingHosts/com.apple.passwordmanager.json /Library/Google/Chrome/NativeMessagingHosts/com.apple.passwordmanager.json
fi
if [ -f "/System/Cryptexes/App/Library/Application Support/Mozilla/NativeMessagingHosts/com.apple.passwordmanager.json" ]; then
    mkdir -p "/Library/Application Support/Mozilla/NativeMessagingHosts"
    ln -sf "/System/Cryptexes/App/Library/Application Support/Mozilla/NativeMessagingHosts/com.apple.passwordmanager.json" "/Library/Application Support/Mozilla/NativeMessagingHosts/com.apple.passwordmanager.json"
fi
echo "Removing the cryptex payload"
/bin/rm -rfxv "$cryptex_dir" || exit 4
exit $install_status
```

Shell script — 36 lines

The screenshot shows the Mac OS X Package Contents browser interface. The title bar reads "Safari18.1SonomaManualSeed.pkg". The left sidebar lists package contents under "Safari18.1Sonoma.pkg", including sections for preinstall, postinstall, and Tools. The main pane displays the "graftCryptex.sh" shell script, which handles the installation and removal of Cryptex payloads for different applications. A large yellow emoji with a neutral face is overlaid on the script. On the right, the text "Ability to revert?" is displayed in large blue letters.

```
#!/bin/sh
targetVolume="$3"
cryptegraft="$targetVolume/System/Library/PrivateFrameworks/MobileSoftwareUpdate.framework/Support/cryptegraft"
cryptex_dir="$targetVolume/Library/Apple/Safari/Cryptex"
if [ ! -f "$cryptegraft" ] || [ ! -x "$cryptegraft" ]; then
    echo "Unable to find and execute '$cryptegraft'"
    exit 1
fi
if [ ! -d "$cryptex_dir" ]; then
    echo "The Cryptex directory does not exist"
    exit 2
fi
echo "Installing the Cryptex"
"$cryptegraft" --downlevel --updateBundle "$cryptex_dir" --Restore --targetVolume "$targetVolume"
install_status=$?
# rdar://110673814
if [ -f "/System/Cryptexes/App/Library/Google/Chrome/NativeMessagingHosts/com.apple.passwordmanager.json" ]; then
    mkdir -p "/Library/Google/Chrome/NativeMessagingHosts"
    ln -sf "/System/Cryptexes/App/Library/Google/Chrome/NativeMessagingHosts/com.apple.passwordmanager.json" "/Library/Google/Chrome/NativeMessagingHosts/com.apple.passwordmanager.json"
fi
if [ -f "/System/Cryptexes/App/Library/Application Support/Mozilla/NativeMessagingHosts/com.apple.passwordmanager.json" ]; then
    mkdir -p "/Library/Application Support/Mozilla/NativeMessagingHosts"
    ln -sf "/System/Cryptexes/App/Library/Application Support/Mozilla/NativeMessagingHosts/com.apple.passwordmanager.json" "/Library/Application Support/Mozilla/NativeMessagingHosts/com.apple.passwordmanager.json"
fi
echo "Removing the cryptex payload"
/bin/rm -rf xv "$cryptex_dir" || exit 4
exit $install_status
```



# Private Cloud Compute



# Private Cloud Compute



## Standard & Codeless Cryptexes

- Debug Shell Cryptex
- Model Weight Cryptex (data only)



# Private Cloud Compute



## Standard & Codeless Cryptexes

- Debug Shell Cryptex
- Model Weight Cryptex (data only)

Please read the Private Cloud Compute Documentation, really neat stuff:  
<https://security.apple.com/documentation/private-cloud-compute/>

**RSR technologies live on** 

**Even if not directly through “Rapid Security Response” updates**

# **“Is Rapid Security Response a Failure?”**

**- Me, complaining on the internet (April, 2023)**

**“Is Rapid Security Response a Failure?”**

**macOS 15.1.1 & iOS 18.1.1...**

# macOS 15.1.1 & iOS 18.1.1...

A screenshot of a web browser window displaying the macOS 15.1.1 update information from support.apple.com. The title is "macOS Sequoia 15.1.1". It was released on November 19, 2024. The "JavaScriptCore" section lists it as available for macOS Sequoia, impacting Intel-based Mac systems with the potential for arbitrary code execution if exploited. The "WebKit" section also lists it as available for macOS Sequoia, impacting Intel-based Mac systems with the potential for a cross site scripting attack if exploited. Both sections mention a cookie management issue was addressed with improved state management. WebKit Bugzilla reference numbers 283063 and 283095 are provided, along with CVE-2024-44308 and CVE-2024-44309.

**macOS Sequoia 15.1.1**

Released November 19, 2024

**JavaScriptCore**

Available for: macOS Sequoia

Impact: Processing maliciously crafted web content may lead to arbitrary code execution. Apple is aware of a report that this issue may have been actively exploited on Intel-based Mac systems.

Description: The issue was addressed with improved checks.

WebKit Bugzilla: 283063

CVE-2024-44308: Clément Lecigne and Benoît Sevens of Google's Threat Analysis Group

**WebKit**

Available for: macOS Sequoia

Impact: Processing maliciously crafted web content may lead to a cross site scripting attack. Apple is aware of a report that this issue may have been actively exploited on Intel-based Mac systems.

Description: A cookie management issue was addressed with improved state management.

WebKit Bugzilla: 283095

CVE-2024-44309: Clément Lecigne and Benoît Sevens of Google's Threat Analysis Group

A screenshot of a web browser window displaying the iOS 18.1.1 update information from support.apple.com. The title is "iOS 18.1.1 and iPadOS 18.1.1". It was released on November 19, 2024. The "JavaScriptCore" section lists it as available for various iOS and iPadOS devices, impacting them with the potential for arbitrary code execution if exploited. The "WebKit" section also lists it as available for these devices, impacting them with the potential for a cross site scripting attack if exploited. Both sections mention a cookie management issue was addressed with improved state management. WebKit Bugzilla reference number 283063 is provided, along with CVE-2024-44308 and CVE-2024-44309.

**iOS 18.1.1 and iPadOS 18.1.1**

Released November 19, 2024

**JavaScriptCore**

Available for: iPhone XS and later, iPad Pro 13-inch, iPad Pro 12.9-inch 3rd generation and later, iPad Pro 11-inch 1st generation and later, iPad Air 3rd generation and later, iPad 7th generation and later, and iPad mini 5th generation and later

Impact: Processing maliciously crafted web content may lead to arbitrary code execution. Apple is aware of a report that this issue may have been actively exploited on Intel-based Mac systems.

Description: The issue was addressed with improved checks.

WebKit Bugzilla: 283063

CVE-2024-44308: Clément Lecigne and Benoît Sevens of Google's Threat Analysis Group

**WebKit**

Available for: iPhone XS and later, iPad Pro 13-inch, iPad Pro 12.9-inch 3rd generation and later, iPad Pro 11-inch 1st generation and later, iPad Air 3rd generation and later, iPad 7th generation and later, and iPad mini 5th generation and later

Impact: Processing maliciously crafted web content may lead to a cross site scripting attack. Apple is aware of a report that this issue may have been actively exploited on Intel-based Mac systems.

Description: A cookie management issue was addressed with improved state management.

WebKit Bugzilla: 283095

CVE-2024-44309: Clément Lecigne and Benoît Sevens of Google's Threat Analysis Group

# macOS 15.1.1 & iOS 18.1.1...

**IPSWs**

- iPhone17,1\_18.1\_22B82\_Restore.ipsw
- iPhone17,1\_18.1.1\_22B91\_Restore.ipsw

**Kernel**

**Version**

iOS	Version	Build	Date
18.1 (22B82)	24.1.0	11215.42.1~1	Mon, 07Oct2024 21:14:29 PDT
18.1.1 (22B91)	24.1.0	11215.42.1~1	Mon, 07Oct2024 21:14:29 PDT

**MachO**

Updated (1)

View Updated

Entitlements

Entitlements DIFF

**DSC**

**WebKit**

iOS	Version
18.1 (22B82)	619.2.8.10.7
18.1.1 (22B91)	619.2.8.10.9

Userspace-only fixes...

Thank you blacktop\_ for the diff!

# macOS 15.1.1 & iOS 18.1.1...

The screenshot shows a GitHub diff interface comparing two versions of iOS. The top navigation bar indicates the URL is github.com. The main content area is titled "18.1 (22B82) .vs 18.1.1 (22B91)".

- IPSWs**:
  - iPhone17,1\_18.1\_22B82\_Restore.ipsw
  - iPhone17,1\_18.1.1\_22B91\_Restore.ipsw
- Kernel**:

iOS	Version	Build	Date
18.1 (22B82)	24.1.0	11215.42.1~1	Mon, 07Oct2024 21:14:29 PDT
18.1.1 (22B91)	24.1.0	11215.42.1~1	Mon, 07Oct2024 21:14:29 PDT
- MachO**:
  - Updated (1)
  - [View Updated](#)
- Entitlements**:
  - [Entitlements DIFF](#)
- DSC**
- WebKit**:

iOS	Version
18.1 (22B82)	619.2.8.10.7
18.1.1 (22B91)	619.2.8.10.9

Userspace-only fixes...

This ~~meeting~~ security update  
could've been ~~an email~~ a  
Rapid Security Response

Thank you blacktop\_ for the diff!

# Conclusion



I JUST THINK THEY'RE NEAT!

# Thanks for listening to my rambles!

Mirrored on [khronokernel.com](http://khronokernel.com)



## Socials

- Twitter: <https://twitter.com/khronokernel>
- GitHub: <https://github.com/khronokernel>
- LinkedIn: <https://www.linkedin.com/in/mykola-grymalyuk>

(If you find my work interesting, I'd would love the opportunity to work more in this field!)

Credits to those who directly/indirectly helped make this possible!

- Blacktop\_: IPSW Diffs
- MrMacintosh: Archival of Software Updates
- ASentientBot & DhinakG: Reverse Engineering Cryptexes & OpenCore Legacy Patcher work



# References

References:

- Security Research Device:
  - [archive.org](https://archive.org) (August 3rd, 2020): <https://developer.apple.com/programs/security-research-device/>
- CVE-2023-37450:
  - Security Contents: <https://support.apple.com/en-il/106354>
- Aaron's report on 13.4.1 (a) being pulled:
  - Twitter: <https://x.com/aaronp613/status/1678603028113289217>
- Private Cloud Compute Cryptex Usage:
  - Software Layering: <https://security.apple.com/documentation/private-cloud-compute/softwarelayering#Cryptexes>
- macOS 15.1.1 Security Contents:
  - <https://support.apple.com/en-ca/121753>
- iOS 18.1.1 Security Contents:
  - <https://support.apple.com/en-ca/121752>
- Blacktop\_’s iOS 18.1.1 diff against 18.1:
  - [https://github.com/blacktop/ipsw-diffs/blob/main/18\\_1\\_22B82\\_vs\\_18\\_1\\_1\\_22B91/README.md](https://github.com/blacktop/ipsw-diffs/blob/main/18_1_22B82_vs_18_1_1_22B91/README.md)
- macOS 13.4.1 (c) User agent:
  - <https://tidbits.com/2023/07/13/rapid-security-responses-for-ios-ipados-16-5-1-a-and-macos-ventura-13-4-1-a-2/>
- macOS 13.4.1 IPSW:
  - [https://updates.cdn-apple.com/2023SpringFCS/fullrestores/042-01877/2F49A9FE-7033-41D0-9DOC-64EFCE6B4C22/UniversalMac\\_13.4.1\\_22F82\\_Restore.ipsw](https://updates.cdn-apple.com/2023SpringFCS/fullrestores/042-01877/2F49A9FE-7033-41D0-9DOC-64EFCE6B4C22/UniversalMac_13.4.1_22F82_Restore.ipsw)
- macOS 13.4.1 (a) RSR payload:
  - x86\_64: <https://updates.cdn-apple.com/2023SummerFCS/patches/042-11156/A193B5BA-AC1C-4074-A4EB-18BE0E10DBAD/com.apple.MobileAsset.MacSplatSoftwareUpdate/d89d81737950136e2d6106ecfbff16aa024e8e6.zip>
  - arm64: <https://updates.cdn-apple.com/2023SummerFCS/patches/042-11155/2C86EOCF-DB4F-4C20-8925-2F3F54F61A11/com.apple.MobileAsset.MacSplatSoftwareUpdate/fda10f2f66899a3530fd1cc7e99d0267eabef6c2.zip>
- Shameless plug:
  - <https://khronokernel.com/macos/2023/04/18/RSR.html>