macOS Cuckoo Stealer | Ensuring Detection and Defense as New Samples Rapidly Emerge

By Phil Stokes

Published: 2024-05-09 · Archived: 2026-04-05 17:06:52 UTC

Infostealers targeting macOS devices have been on the rise for well over a year now, with variants such as Atomic Stealer (Amos), RealStealer (Realst), MetaStealer and others widely distributed in the wild through malicious websites, cracked applications and trojan installers. These past few weeks have seen a new macOS malware family appear that researchers have dubbed 'Cuckoo Stealer', drawing attention to its abilities to act both as an infostealer and as spyware.

In this post, we review Cuckoo Stealer's main features and logic from a detection point of view and offer extended indicators of compromise to aid threat hunters and defenders. At the time of writing the latest version of XProtect, version 2194, does not block execution of Cuckoo Stealer malware. SentinelOne customers are protected from macOS Cuckoo Stealer.

More Cuckoo Stealers Appearing

Since the initial report on the emergence of this family of malware on April 30, we have seen a rise in new samples and trojanized applications from the four originally reported by Kandji to 18 unique trojanized applications at the time of writing, with new samples appearing daily.

The trojanized apps are various kinds of "potentially unwanted programs" offering dubious services such as PDF or music converters, cleaners and uninstallers (a full list appears in the IoCs at the end of this post) such as:

- App Uninstaller.app
- DumpMedia Amazon Music Converter.app

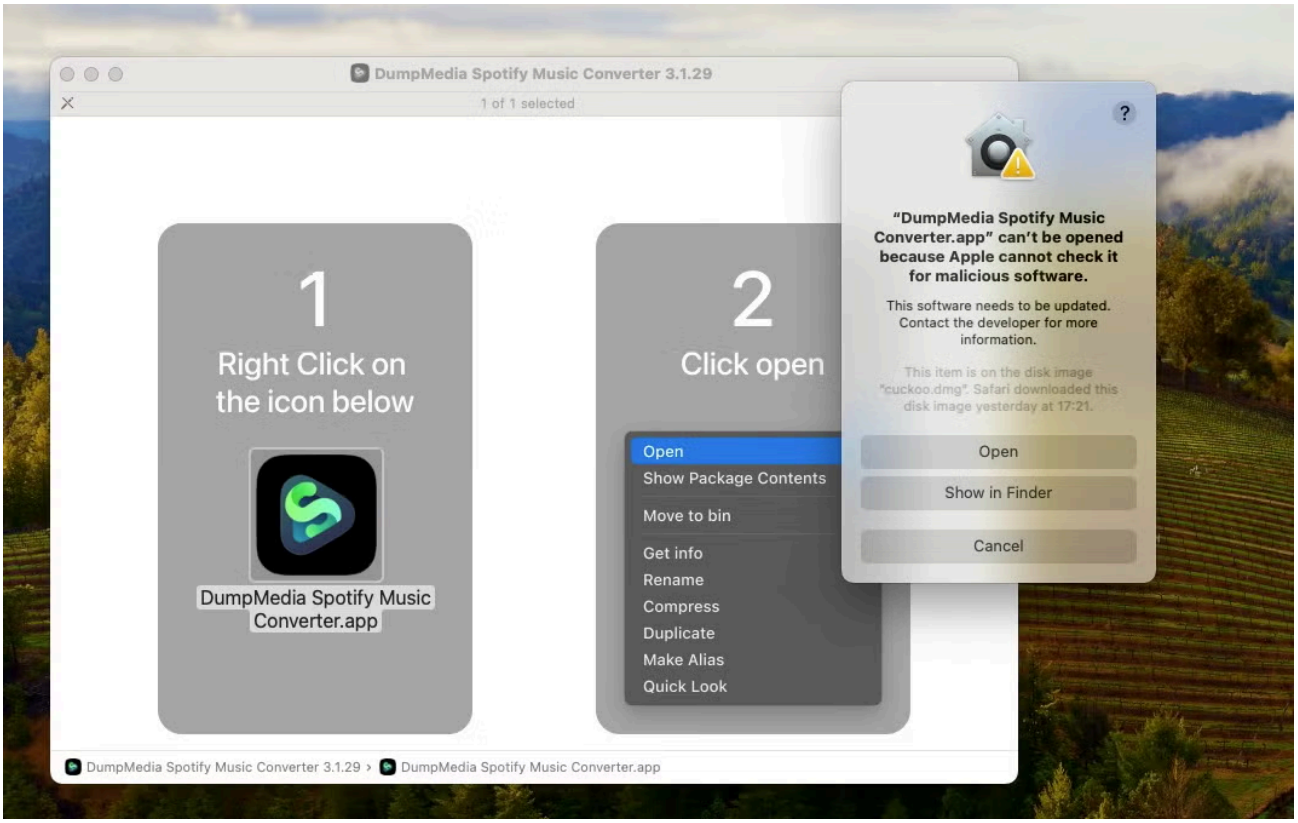
- FoneDog Toolkit for Android on Mac.app
- iMyMac PDF Compressor.app
- PowerUninstall.app
- TuneSolo Apple Music Converter.app

As reported previously, these applications contain a malicious binary in the MacOS folder named `upd`. The most recent binaries – in 'fat' and 'thin' versions for both Intel x86 and arm64 architectures – are *ad hoc* codesigned and their parent applications all share the same bundle identifier, `upd.upd`.

Apple's `codesign` utility will provide identical output for all these samples:

```
codesign -dv file
...
Identifier=upd.upd
Format=Mach-O thin (x86_64)
CodeDirectory v=20400 size=1536 flags=0x2(adhoc) hashes=38+7 location=embedded
Signature=adhoc
Info.plist=not bound
TeamIdentifier=not set
Sealed Resources=none
Internal requirements count=0 size=12
```

Some protection is offered to unsuspecting users by Apple's Gatekeeper, which will by default throw a warning that the application is not notarized. The malware authors have anticipated this and provided the user with instructions on how to run the application.



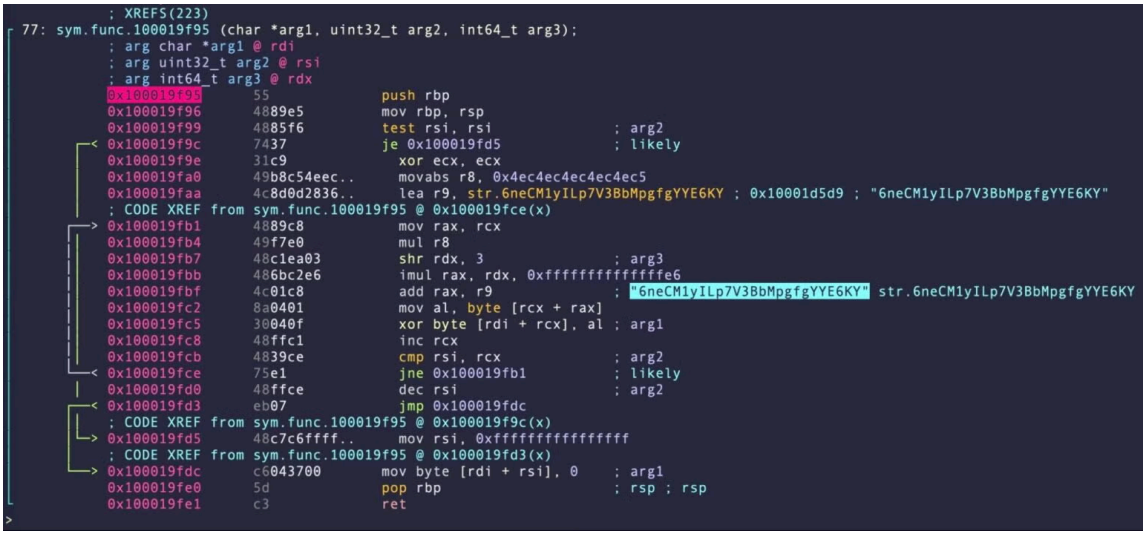
The malware is written in C++ and was created in build 12B45b of Xcode, version 12.2, a rather old version that was released in November 2020, using a device still running macOS 11 Big Sur (build 20A2408) from the same year.

The code signature and the application's `Info.plist` containing this information make current samples relatively easy to identify.

Simple Obfuscation Helps Cuckoo to Hide in Apple's Nest

A noticeable characteristic of the malware is the heavy use of XOR'd strings in an attempt to hide its behavior from simple static signature scanners. The samples use different XOR keys (see the list of IoCs at the end of this post) of varying lengths to decrypt the main strings and functionality dynamically.

Though the binary is stripped and lacks function names, the decrypt routine is readily identifiable from the large number of cross references to it in the rest of the code. Current samples call the decrypt routine precisely 223 times.



Cuckoo decryption function

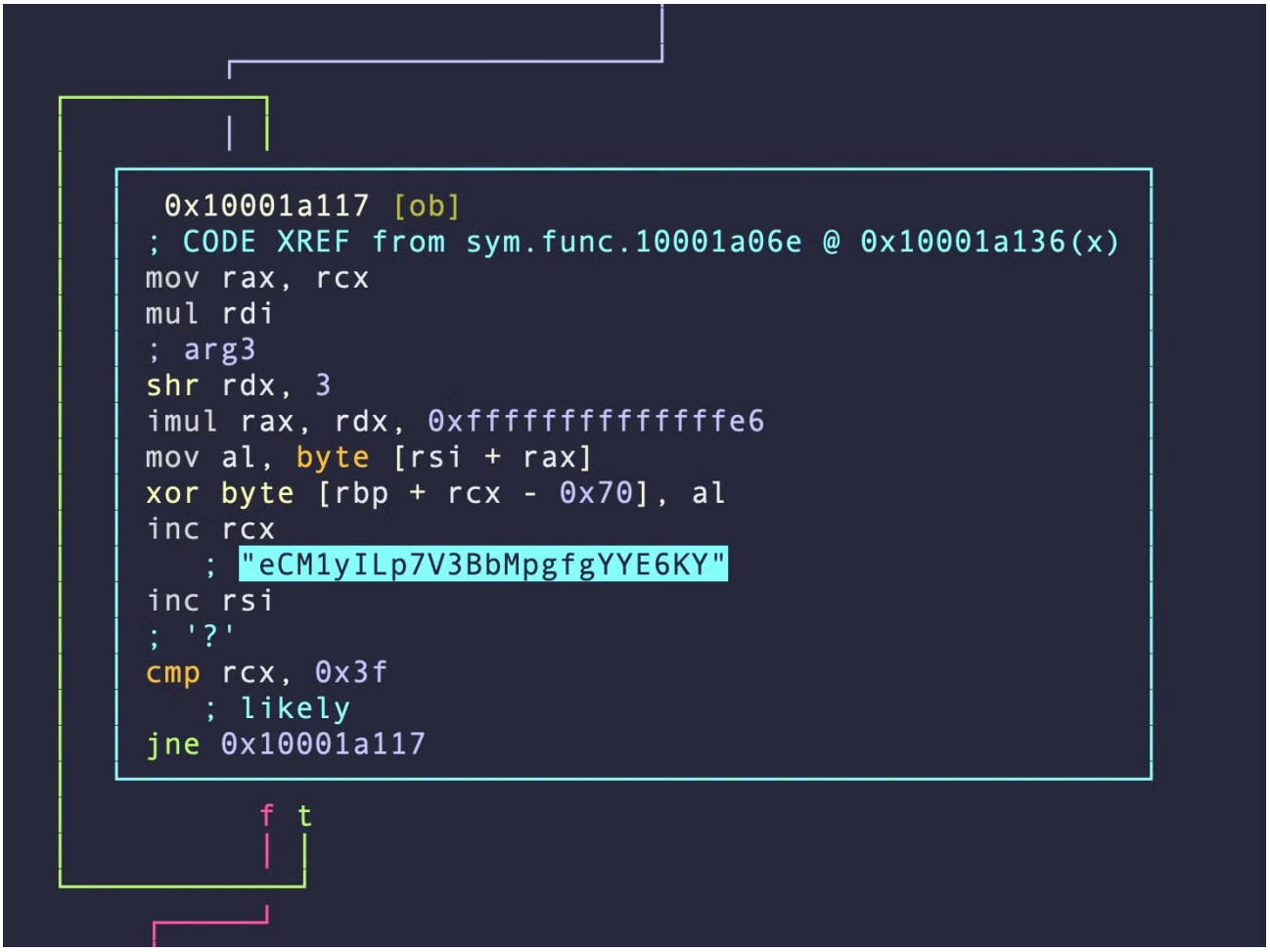
By breaking on this function in a debugger, it is relatively straightforward to output the decrypted strings to understand the malware's behavior.

However, not all obfuscated strings are processed through this function. The decryption key and routine can be found independently in other places in the code as well.

Of the few unobfuscated strings in the current binary is one that represents an array of file extensions, indicating the kind of information the malware authors are interested in stealing.

```
{"txt", "rtf", "doc", "docx", "xls", "xlsx", "key", "wallet", "jpg", "dat", "pdf", "pem", "asc", "ppk", "r
```

Looking for cross references to 'wallet' (one of the items in the array), we find the array is consumed in a function which calls both the decrypt function and another function that implements the same XOR routine and key.

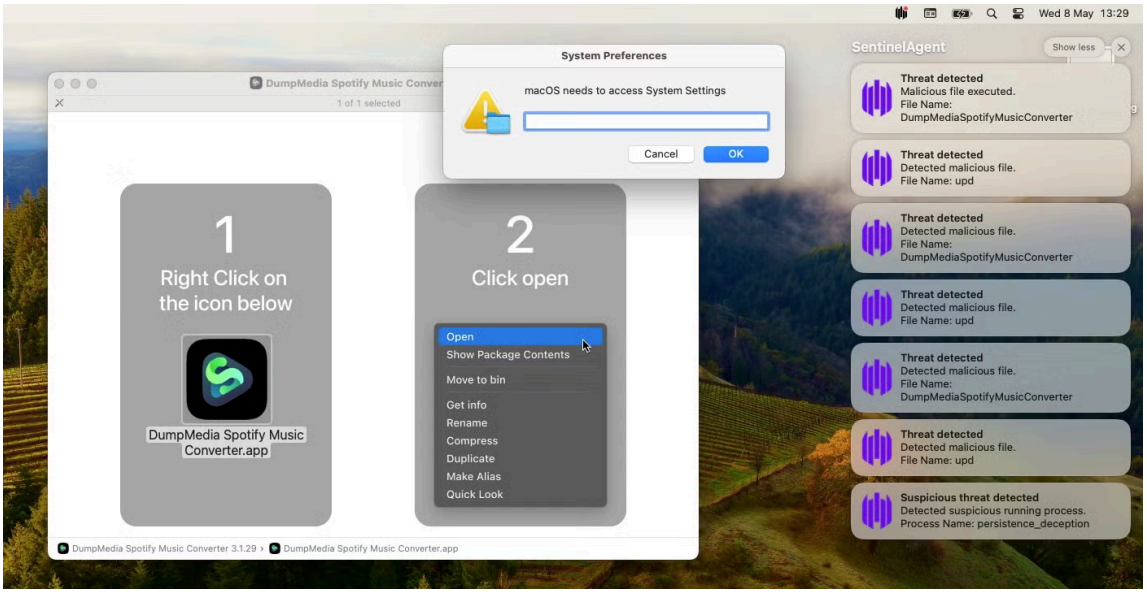


In radare2, we can find all references to the XOR key via `grepping` the output of the `ax` command for the string's address.

Finding cross references in radare2

Cuckoo Stealer Observable Behavior

Despite these attempts at obfuscation, analysis of Cuckoo Stealer reveals that, unsurprisingly, it uses many of the same techniques as other infostealers we have encountered in the last 12 months or so. In particular, it makes various uses of AppleScript to duplicate files and folders of interest and to steal the user's admin password in plain text.



SentinelOne detects Cuckoo Stealer

This is achieved through a simple AppleScript dialog using the “hidden answer” option, a ploy that macOS attackers have been using since at least 2008, as we observed recently in relation to Atomic Stealer.

> Erm #Microsoft , OK, but has #Apple security also failed?
> In 2008, before #XProtect, in OS X 10.5 Leopard, PokerStealer was scraping passwords in clear text same way as Atomic Stealer, MetaStealer and others are still doing in 2024.
> fa91b42b68d92f57b56929cb35c12ae54e022ad2 pic.twitter.com/oe2MNHJLsf
>
> — Phil Stokes ⌐◨-◨¬ (@philofishal) April 4, 2024

With Cuckoo Stealer, if the user enters anything other than a valid admin password, the malware will repeatedly display the dialog until the right password is provided. This remains true even if the user presses the ‘Cancel’ button.

The underlying mechanism for how the password is checked was nicely elucidated by Kandji researchers here. The scraped password is then saved in clear text in a file named `pw.dat` in a hidden subfolder of the User’s home directory. The hidden folder’s name is a combination of `.local-` and a randomly generated `UUID` identifier. For example:

```
~/.local-6635DD81-94DD-59E3-9D84-20BD41C51999/
```

The following regexes can be used to find paths or commands containing this pattern:

```
\.local-[[:xdigit:]]{8}-[[:xdigit:]]{4}-[[:xdigit:]]{4}-[[:xdigit:]]{4}-[[:xdigit:]]{12}/

// alternatively:
\.local-[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}/
```

In addition, the malware also attempts to install a persistence LaunchAgent with the label `com.user.loginscript`. The name of the property list file itself will take the form of the parent application bundle. For example, the trojan DumpMedia Spotify Music Converter.app will create a plist called

`~/Library/LaunchAgents/com.dumpmedia.spotifymusicconverter.plist`, while iMyMac Video Converter.app will write the same plist out as `com.immyac.videoconverter.plist`.

com.imymac.videoconverter.plist

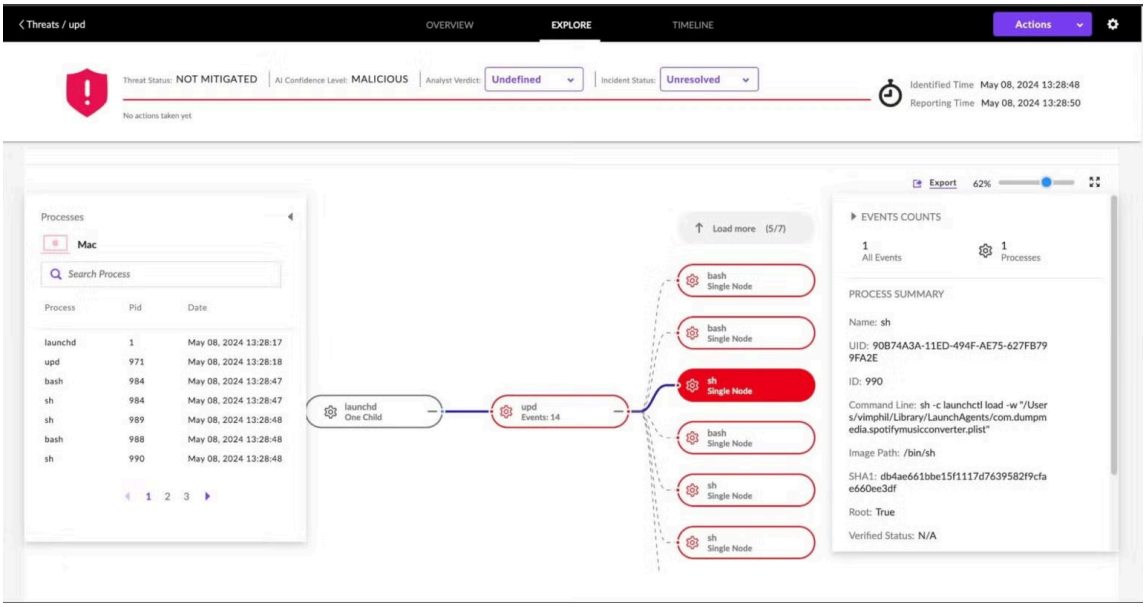
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
   <key>Label</key>
   <string>com.user.loginscript</string>
   <key>ProgramArguments</key>
   <array>
      <string>/Users/vimphil/.local-6635DD81-94DD-59E3-9D84-20BD41C51999/iMyMacVideoConverter</string>
   </array>
   <key>StartInterval</key>
   <integer>60</integer>
</dict>
</plist>
```

Cuckoo Stealer LaunchAgent

This persistence agent will point to a copy of the `upd` binary located in the same hidden `.local-<UUID>` directory mentioned above.

The malware also makes use of several Living Off the Land utilities including `xattr`, `osascript` and `system_profiler` for discovery.

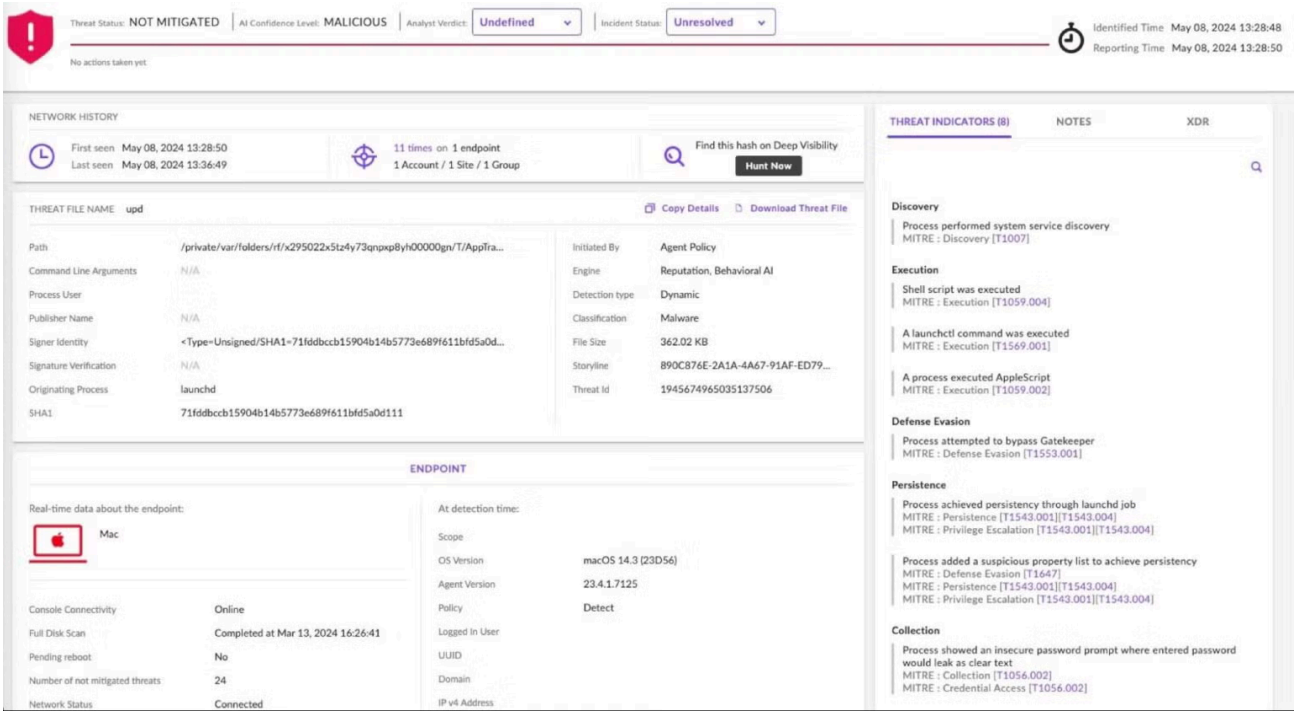
Command	Arguments
awk	/Hardware UUID/{print $(NF)}
launchctl	load -w “/Users/user1/Library/LaunchAgents/com.dumpmedia.spotifymusicconverter.plist”
osascript	-e ‘display dialog “macOS needs to access System Settings” default answer “” with title “System Preferences” with icon caution with hidden answer’
system_profiler	SPHardwareDataType | awk ‘/Hardware UUID/{print $(NF)}’
xattr	-d com.apple.quarantine “/Users/user1/.local-6635DD81-94DD-59E3-9D84-20BD41C51999/DumpMediaSpotifyMusicConverter”



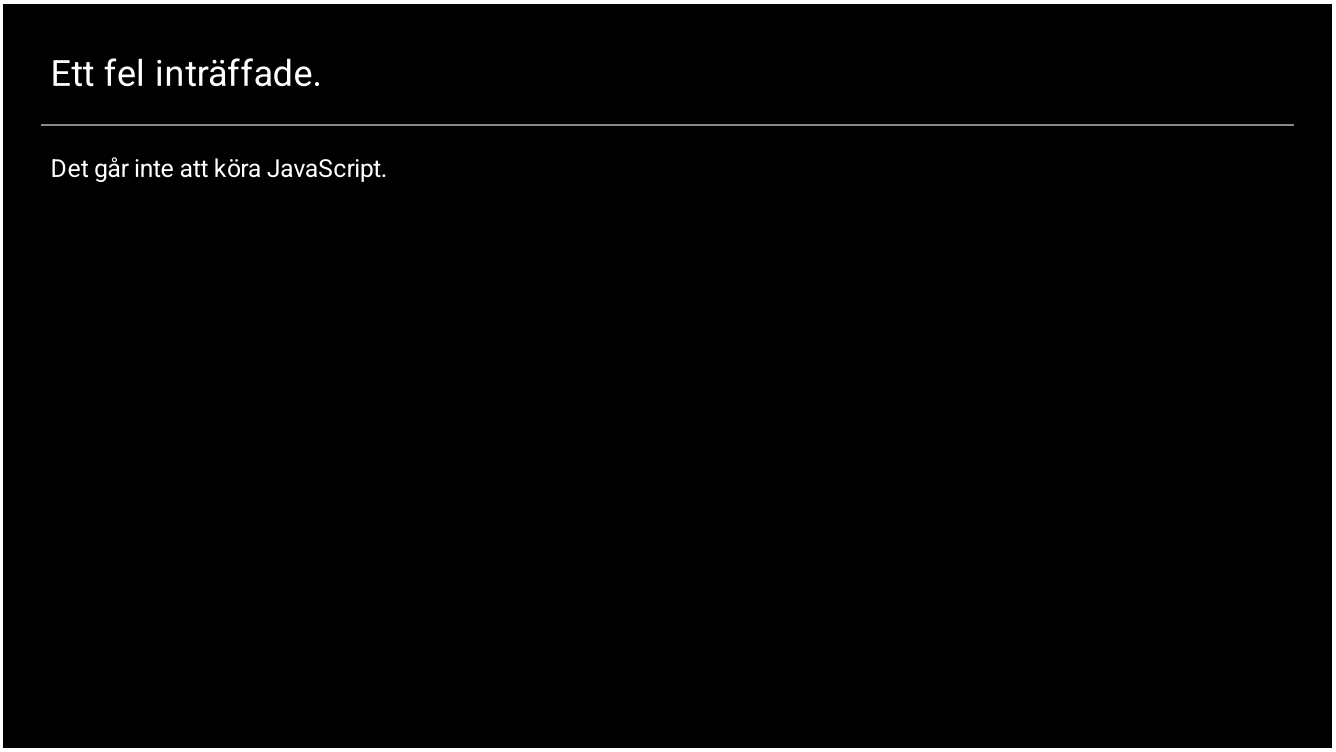
Cuckoo Stealer execution chain

SentinelOne Protects Against Cuckoo Stealer

SentinelOne Singularity detects Cuckoo Stealer and prevents its execution when the policy is set to Protect/Protect. In Detect mode, the agent will allow analysts to observe and investigate malicious behavior, as shown below.



Agent version 23.4.1.7125 and later offer an extensive set of behavioral indicators including reference to MITRE TTPs specific to macOS infostealers.



Conclusion

The actors behind the Cuckoo Stealer campaign have clearly invested some resources into developing a novel infostealer rather than buying any of the ready-made offerings currently circulating in various Telegram channels and darknet forums. This, along with the rising numbers of samples we have observed since initial reporting of this threat, suggests that we will likely see further variants of this malware in the future.

Enterprises are advised to use a third party security solution such as SentinelOne Singularity to ensure that devices are protected against this and other threats targeting macOS devices in the fleet.

To learn more about how SentinelOne can help protect your organization, contact us or request a free demo.

Indicators of Compromise

Bundle Identifier

upd.upd

Observed Application Names

App Uninstaller.app
DumpMedia Amazon Music Converter.app
DumpMedia DeezPlus.app
DumpMedia Pandora Music Converter.app
DumpMedia Spotify Music Converter.app
DumpMedia Video Converter.app
DumpMedia YouTube Music Converter.app
FoneDog Data Recovery.app
FoneDog iPhone Cleaner.app
FoneDog PDF Compressor.app
FoneDog Toolkit for Android on Mac.app
FoneDog Toolkit for iOS on Mac.app
FoneDog Video Converter.app
iMyMac PDF Compressor.app
iMyMac Video Converter.app
PowerUninstall.app
TunesFun Apple Music Converter.app
TuneSolo Apple Music Converter.app

Observed Mach-Os (SHA1)

04a572b2a17412bba6c875a43289aac521f7b98d
0e3e58a2b19072823df2ec52f09e51acf0d0d724
127c486eab9398a2f42208d96aa12dd8fcfb68b5
1ef1f94d39931b6e625167b021a718f3cfe6bb80
1f49bb334ebcec6b2493d157caf90a8146fb68d9
219f57e9afe201ad4088340cd5b191223d4c4227
24c311abe5d93d21172a6928ba3a211528aa04f9
266f48c38efbb5a6d49fb74194c74fe68d02d62a

298c9ab225d7262a2106bc7bec0993eaa1210a0d
2a422057790bae755c3225aff3e47977df234b11
2c7ec5358b69f8e36c35c53501e4ba6efce25689
2cdda89c50c2aa1eb4b828350b7086748c58fe08
35d75565de813e89a765718ed31c1bfebfd3c11c
4cf895c391557498d2586cee3ace3c32a3a83a4e
4cfdf872051900df8a959b95a03f6c906ad4596e
50360b325aad398a5d580a2adc9aef597eb98855
5220a53c1930ea93849caa88850cb6628a06cd90
57a1f3d3cbbc33b92177660ee620bff4f1c5b229
63eb1abe69b11c8ae04092ccf822633d1e1ff648
69c6c1f09f8a1ad61f1c48527ff27e56847a716f
6aba0ebabccea1902ba2ab7ac183a4bd22617555
71fddbccb15904b14b5773e689f611bfd5a0d111
82c70c956f5f66cf642991285fd631a9094abbf4
873fd2fc21457e707832c859534d596a7c803a46
8bab36fe676c8296ef3889d5ef0afcc4b3f017f3
8bc02ae4262eaf2cbb2454709db7f95cebcc9432
8bee44d0e4e22d3a85cfb9d00d00cb7d85433c9d
8c10459be56dde03c75cda993a489373a8251abf
9ac058d4541aa0e7ba222d25c55c407451f318a7
9d4b45104b3eb3734cb0ba45ca365b95a4c88505
9efa91a0cba44334b1071344314853699155814f
ac755f6da9877a4fc161d666f866a1d82e6de1b0
ac948abaa90b4f1498e699706407ac0c6d4164c7
b49a69fa41a2d7f5f81dbc2be9ea7cfc45c1f3df
b4bd11aa174d1a2f75aff276a2f9c50c4b6a4a1d
b4da5459ccd0556357f8ccd3471a63eebfa6e3b7
b65880c2aecc15db8afa80f027ed0650be23e8f9
bd5cdf05db06c3a81b0509e9f85c26feb34cea81
c5c8335ed343d14d2150a9ba90e182ca739bde8a
c8a6e4a3b16adf5be7c37b589d36cb2bd9706a92
c98d92e01423800404c77f6f82d62e5e7516d46d
cd04a6df24ab7852267619d388dee17f20c66deb
cf069bcafb6510282c8aeab7282e19abc46d558f
db180e1664e566a3393d884a52b93b35bb33911e
db19034d60973d0bcaa237c24252fe969803bc7c
dfed0ca9d883a45a40b2c23c29557ac4679ef698
e57b537f5f3307c6c59f5477e6320f17a9ba5046
e68f0f0e6102a1cd78d5d32ec7807b2060d08f79
e6fa7fcbaf339df464279b8090f6908fed7b325a
e9180ee202c42e2b94689c7e3fb2532dd5179fad
ecca309e0b43cd7f4517a863b95abf7b89be4584
f4999331606b753daaf6d6ad84917712f1420c85

f6e9081e36ca28bf619aebb40a67c56a2de2806e
fad49cac81011214d7fe3db7fc0bd663ef7bb353

Observed XOR Keys

0dhIscuDmR6xn3VMAG9ZYjBKC4VDeXGbyDyWjHM
4E72G6aXPne5ejcUgAfae6khJB3c871V0QUmkI
6neCM1yILp7V3BbMpgfgYYE6KY
7ricF8bWO0eBNiKEravcj2iIXohSNt
7Y9lGDAyEf9vxEmFgRqpDwYM52NFPbsUc
GXMSjRLvCPrrFnc1xa3xvYd43DfM8
HhvDDxmmfm7QuLH4rP63Fzn2eyW5BzuM3N
Hnyl2YPkOMLTNOndVtQwON
JB3k62Vtqymx09aJtnF9lZrCeIc
JsGqCdROAT1VDpSnxrAyZY45uQvRFP
LydNPzURb22Lxk4fxPkdd
MTGpOAycVm9btlQyEa5xVQPiz
Qmi5gstd6Oc27AJLXJQtEqGMxXzHUx
QssogTgvuTaZzPYZQynw0d
aZeTZw0X2lXM083cgmJQvnmCn9kmt
coOwAdmPtzt5Ps9rvUGOMEeFYajX2nJaismV
rzdbcSkVHXHefChUJQFGjAm12oinXwlyH2sHfiY
vLiOnPSKZ1bqjlp1dwuDvmmeQ3QN

Source: https://www.sentinelone.com/blog/macos-cuckoo-stealer-ensuring-detection-and-defense-as-new-samples-rapidly-emerge/