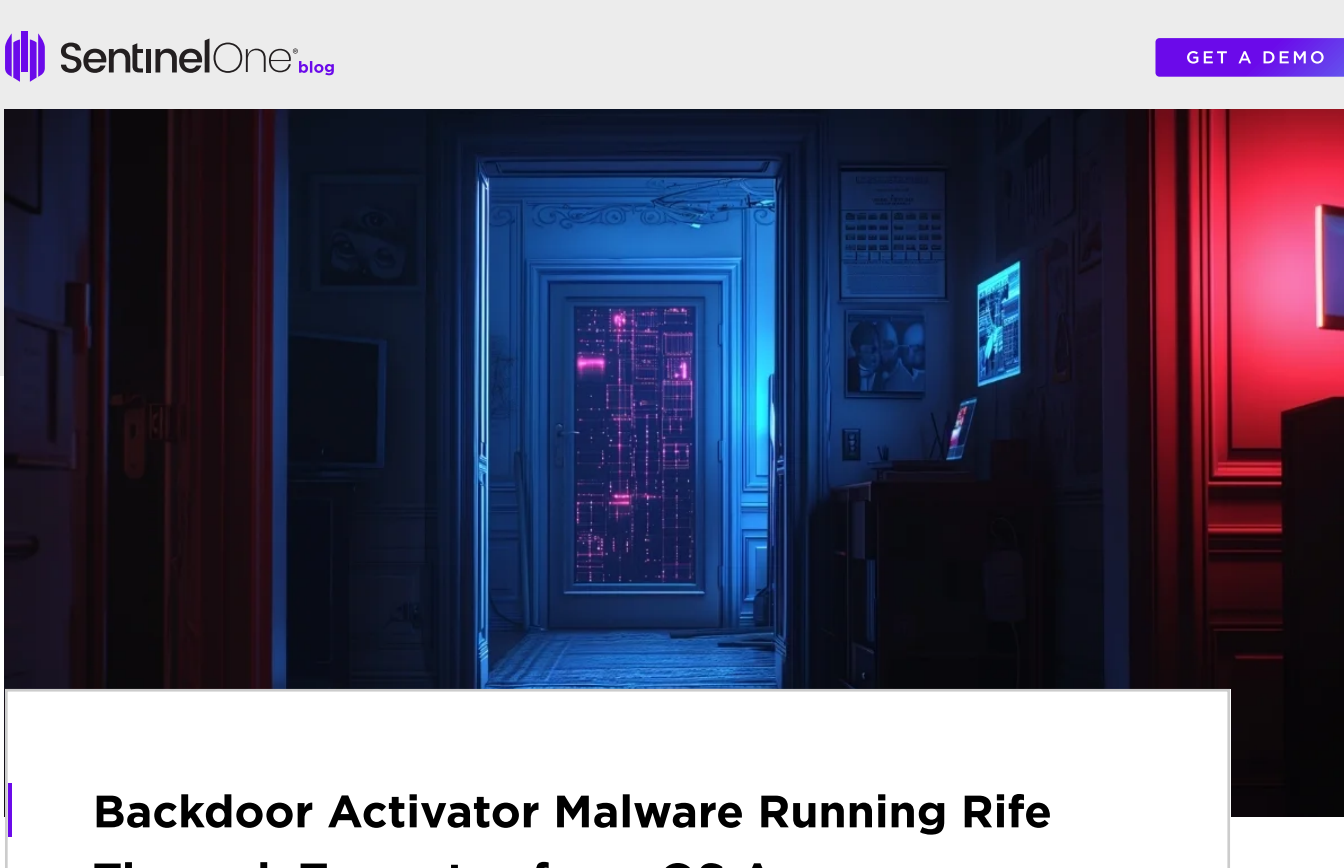
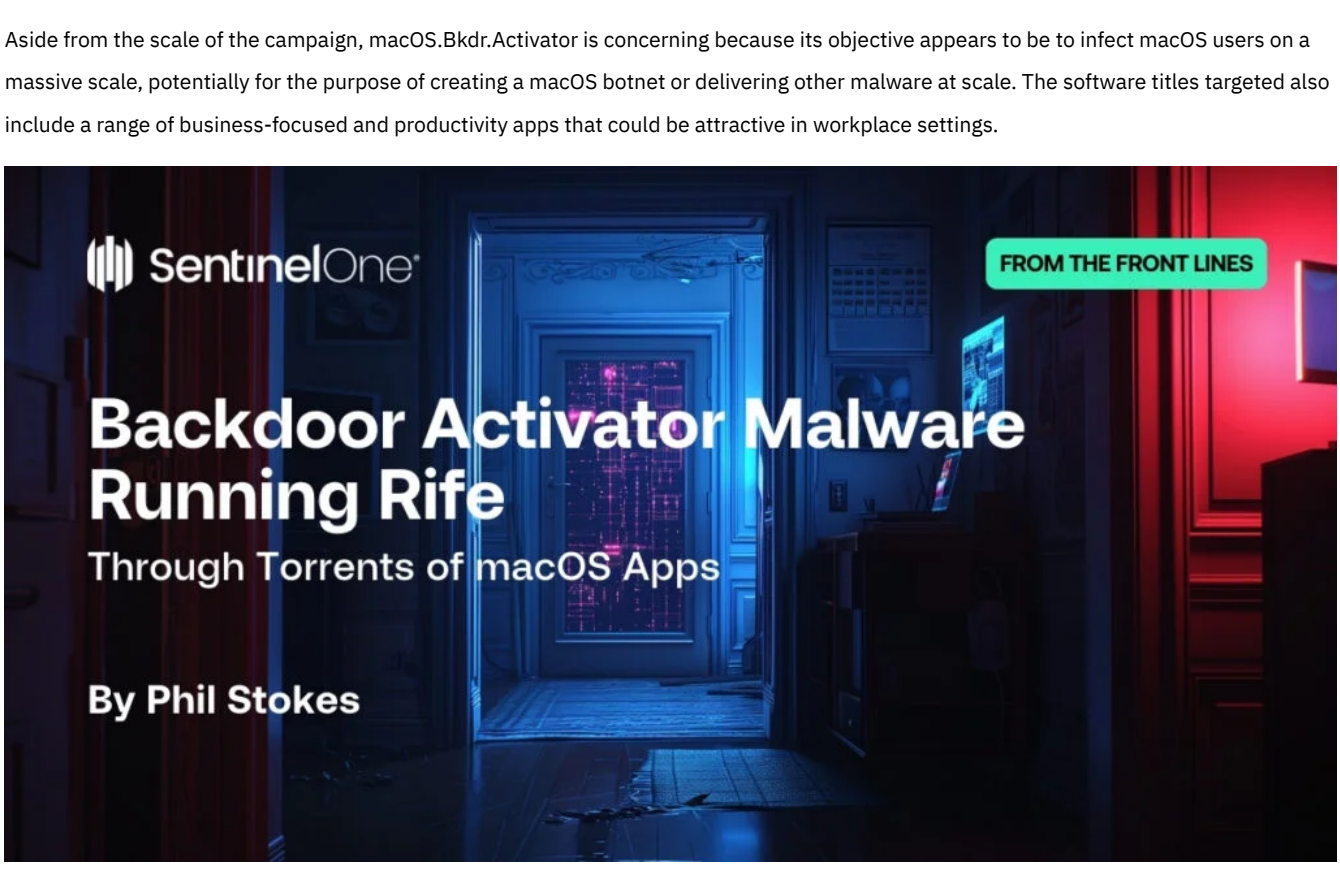


Backdoor Activator Malware Running Rife Through Torrents of macOS Apps

February 1, 2024
by Phil Stokes

Malware authors have long targeted the market for free, cracked apps available through torrent services: in recent years a variety of cryptominers, adware, browser hijackers and bundled software installers have all plied their warez this way, but a recent macOS malware first spotted by researchers at Kaspersky is currently running rampant through dozens of different cracked copies of popular software.

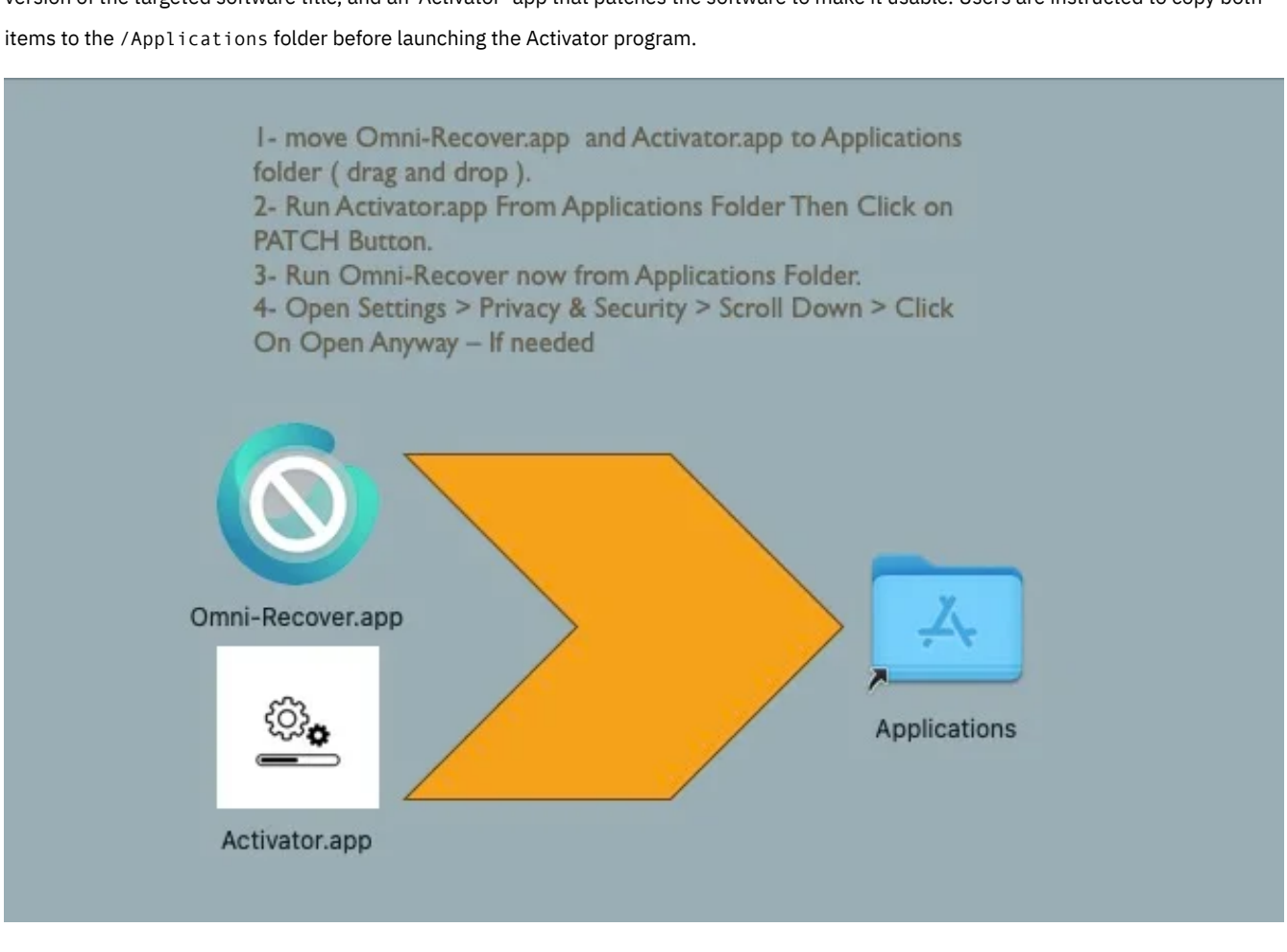
Aside from the scale of the campaign, macOS.Bkdr.Activator is concerning because its objective appears to be to infect macOS users on a massive scale, potentially for the purpose of creating a macOS botnet or delivering other malware at scale. The software titles targeted also include a range of business-focused and productivity apps that could be attractive in workplace settings.



What is macOS.Bkdr.Activator?

Researchers first identified the campaign earlier in January and noted how its multi-stage delivery made use of some novel techniques.

Initial delivery method is via a torrent link which serves a disk image containing two applications: An apparently 'uncracked' and unusable version of the targeted software title, and an 'Activator' app that patches the software to make it usable. Users are instructed to copy both items to the `/Applications` folder before launching the Activator program.

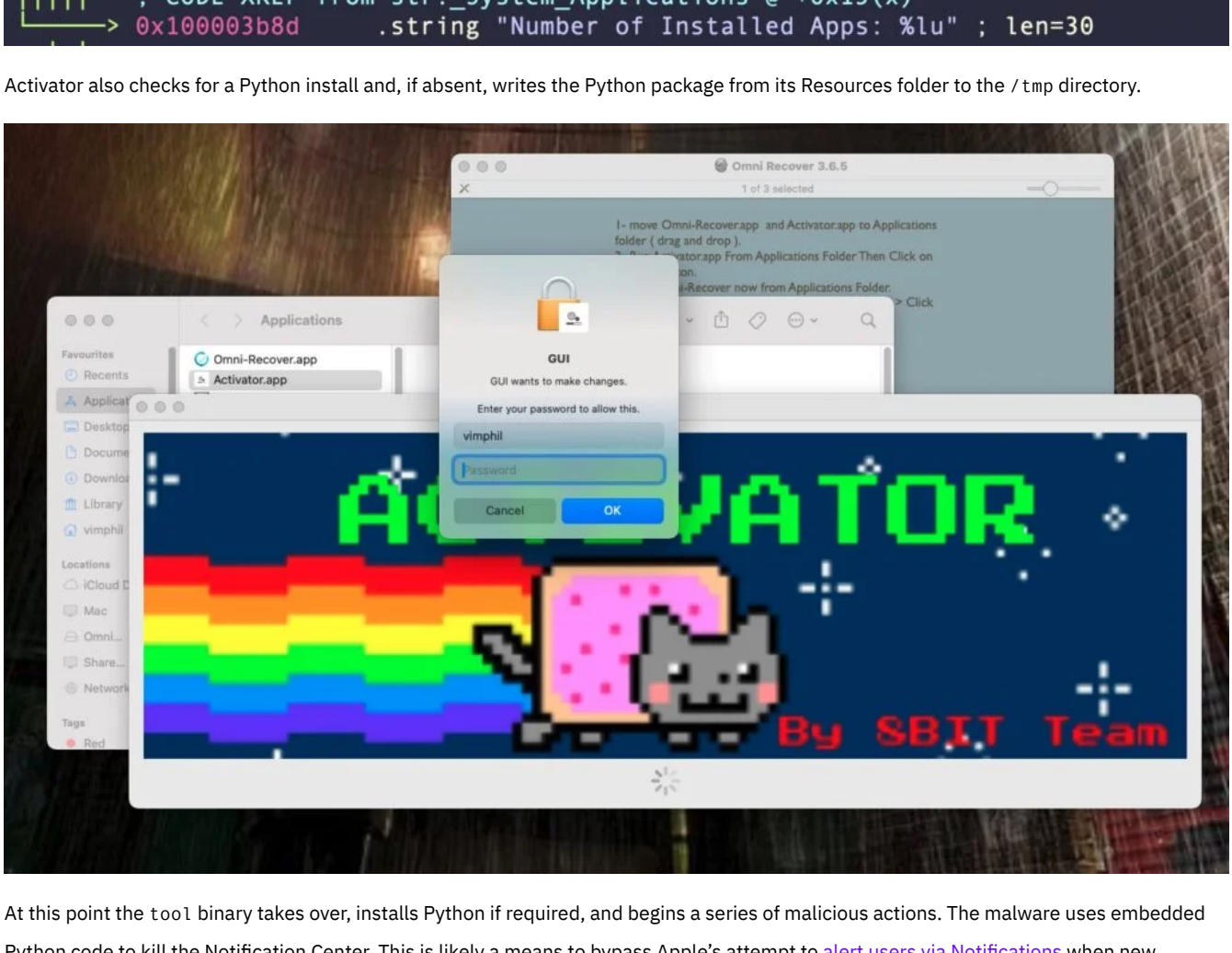


The Activator.app contains two malicious executables: a binary written in Swift named `GUI` located in the bundle's MacOS folder, and a binary written in Objective-C named `tool` and stored in the Resources folder. The latter folder also contains a legitimate, signed installer for Python 3.9.

On launching the Activator.app, victims are asked for an administrator password. This is used to turn off Gatekeeper settings via the `spctl master-disable` command and to allow apps sourced from 'Anywhere' to now run on the device.

```
0x100003b31      .string "Error reading the directory %@: %@" ; len=35
0x100003b50 ~    2540007370        and eax, 0x70730040
;-- str.spctl___master_disable:
;-- hit3_0:
; CODE XREF from str.Error_reading_the_directory___:___ @ +0x4(x)
0x100003b53      .string "spctl --master-disable" ; len=23
;-- str._Applications:
0x100003b6a      .string "/Applications" ; len=14
;-- str._System_Applications:
; CODE XREF from str._Applications @ +0xc(x)
0x100003b78      .string "/System/Applications" ; len=21
;-- str.Number_of_Installed_Apps:__lu:
; CODE XREF from str._System_Applications @ +0x13(x)
0x100003b8d      .string "Number of Installed Apps: %lu" ; len=30
```

Activator also checks for a Python install and, if absent, writes the Python package from its Resources folder to the `/tmp` directory.



At this point the `tool` binary takes over, installs Python if required, and begins a series of malicious actions. The malware uses embedded Python code to kill the Notification Center. This is likely a means to bypass Apple's attempt to alert users via Notifications when new persistence items like LaunchAgents are installed.

```
[0x100003c5e]> s 0x100003928
[0x100003928]> ps
import subprocess\x0d
import time\x0d
\x0d
while True:\x0d
    subprocess.call(['killall', 'NotificationCenter'])\x0d
    time.sleep(0.1)\x0d

[0x100003928]>
```

The Activator contains code to install a LaunchAgent at the following path, where the `%@` variable is replaced with a UUID string generated at runtime.

```
/Library/LaunchAgents/launched.%@.plist
#regex:
/Library/LaunchAgents/launched.[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}\.plist
```

Prior to executing the Python script and installing the LaunchAgent, the `tool` binary attempts to retrieve a remote Python script. If the retrieval is successful, it then leverages the Apple defaults API to determine whether it has ran the same script before. `Defaults` allows programs to store preferences and other information that need to be maintained when the application isn't running. While it is a standard macOS technology, it has rarely been leveraged by malware.

The Activator.app computes a hash of the script and saves it to the user defaults under the key `lastExecutedScriptHash`. If no hash has been previously saved or the stored hash is different, the retrived script is executed.

```
sym._run @ 0x1000023bc(x)
e03740f9         ldr x0, [var_68h]         ; int64_t arg1
bcfcff97         bl sym._hash              ; sym.func.1000016b4 ; sym._hash(0x178158, 0x178998, 0x0)
fd031daa         mov x29, x29              ; sp
e4020094         bl sym.imp.objc_retainAutoreleasedReturnValue ; void objc_retainAutoreleasedReturnValue(void *instance)
                                           ; void objc_retainAutoreleasedReturnValue(0x0000000000000000)
e03300f9         str x0, [sp, 0x60]
2dfdff97     bl sym._readLastExecutedScriptHash ; pc ; sym._readLastExecutedScriptHash(0x0)
fd031daa     mov x29, x29                  ; sp
e0020094     bl sym.imp.objc_retainAutoreleasedReturnValue ; void objc_retainAutoreleasedReturnValue(void *instance)
                                           ; void objc_retainAutoreleasedReturnValue(-1)
e12340f9     ldr x1, [var_40h]
e02f00f9     str x0, [sp, 0x58]
e03340f9     ldr x0, [var_sp_60h]
e22f40f9     ldr x2, [instance]
cd030094     bl sym._objc_msgSend_isEqualToString:
c0000036       tbz w0, 0, 0x10000240c   ; likely
01000014         b 0x1000023fc
sym._run @ 0x1000023f8(x)
000000d0       adrp x0, reloc.objc_msgSend ; 0x100004000
00401291       add x0, x0, 0x490         ; (cstr 0x100003a67) "The hashes are the same." str.cstr.The_hashes_are_the_same.
b5020094       bl sym.imp.NSLog
16000014         b 0x100002460
e12340f9       ldr x1, [var_40h]
e03740f9       ldr x0, [var_68h]
020000d0       adrp x2, reloc.objc_msgSend ; 0x100004000
42c01291       add x2, x2, 0x4b0         ; (cstr 0x100003a80) "#!/usr/bin/env python3" str.cstr.__usr_bin_env_python3
aa030094       bl sym._objc_msgSend_hasPrefix:
e0010036         tbz w0, 0, 0x10000245c ; likely
```

The application's bundle identifier is "-.GUI", so threat hunters may search the defaults database for signs of compromise with:

```
defaults read "-.GUI"
```

macOS Torrents Infected with Backdoor Activator

We have found several hundred unique Mach-O binaries on VirusTotal that are infected with macOS.Bkdr.Activator. Some have very low detection rates, and a few are currently not detected by any VirusTotal engines at all.

	Detections	Size	First seen
2DFCECEDB9E018F6124824BB511579EE428DE2AC2E5B6A7A9B41AC6A7DE17640 No meaningful names macho 64bits	0 / 33	183.25 KB	2024-01-17 22:47:54
00EC97113785EBAB85F0DE5ADE155B5B2619B0F6D841DF8E0C58DEDEB501B331 No meaningful names macho 64bits arm	0 / 57	199.09 KB	2024-01-17 01:52:40
ECAA6F59EBFE6FA3B40D053A7AE8625F763C40D97116244FE80A54B823E963B1 No meaningful names macho 64bits	0 / 43	183.25 KB	2024-01-19 03:52:41

Although the following list cannot be considered complete as new samples continue to be found, the malicious binaries we have discovered pertain to over 70 individual 'cracked' apps that have been hijacked for the Activator campaign.

Any of the following applications that have been sourced from a torrent site or anywhere other than their official distribution channels should be considered as a possible indicator of compromise and the host device inspected for signs of malware infection.

4K Video Downloader 1.4.0	4K YouTube to MP3 Pro 5.1.0	Aiseesoft Blu-ray Player	Alarm Clock Pro 15.6
AnyMP4 iOS Cleaner 1.0.30	Battery Indicator 2.17.0	Bike 1.18.0	Boxy SVG 4.21.1
Chain Timer 10.0	Clipsy Clipboard Manager2.1	ColorWell 7.4.1	Cookie 7.2.1
Cover Desk 1.7	DaisyDisk 4.26 (4.26)	DeliverExpress 2.7.11	Disk Xray 4.1.4
Dropshare 5.45	Easy Data Transform 1.46.1	Eon Timer 2.9.11	Final Draft 12.0.10
Fix My iPhone 2.4.9	FonePaw iOS Transfer 6.0.0	FontLab 8.3.0.8766.0 Beta	Fork 2.38
ForkLift 4.0.6	getIRC – IRC Client 1.5	Ghost Buster Pro 2.5.0	GrandTotal 8.2.2
Hides 5.9.2	HitPaw Video Converter 3.3.0	Infuse Pro 7.6.6	Invisible 2.8.0
Iris 1.6.4	iShowUInstantAdvanced 1.4.19	iTubeGo 7.4.0 Cracked	Keep It 2.3.7
MacX DVD Ripper Pro 6.8.2	MacX MediaTrans 7.9	Magic Battery 8.1.1	Magic Disk Cleaner 2.6.0
MarsEdit 5.1.2	MetaImage 2.6.3	Millumin 4 v4.18.d	Mission Control Plus 1.23
Money Pro 2.10.4	MouseBoost Pro 3.3.5	NetWorker Pro 9.0.1	Nisus Writer Express 4.4
Omni Toolbox 1.5.1	OmniFocus Pro 4.0.3	OmniReader Pro 2.6.8	Pastebot 2.4.6
Perfectly Clear 4.6.0.2629	Privatus 7.0.2	QuickLinks 3.2	RAW Power 3.4.17 Cracked
Rhino-8	SimpleMind Pro 2.3.0	SiteSucker Pro 5.3.0	Soulver 3.10.0
SpamSieve 3.0.3	Swinsian 3.0	SyncBird Pro 4.0.8	TechSmith Snagit 2023.2.6
uDock 4.0.3	Unclutter 2.2.6	Valentina Studio Pro 13.7.0	Web Confidential 5.4.3
WiFiSpoof 3.9.3	Xliff Editor 2.9.15	xScope 4.7.0	zFuse Pro 1.7.36

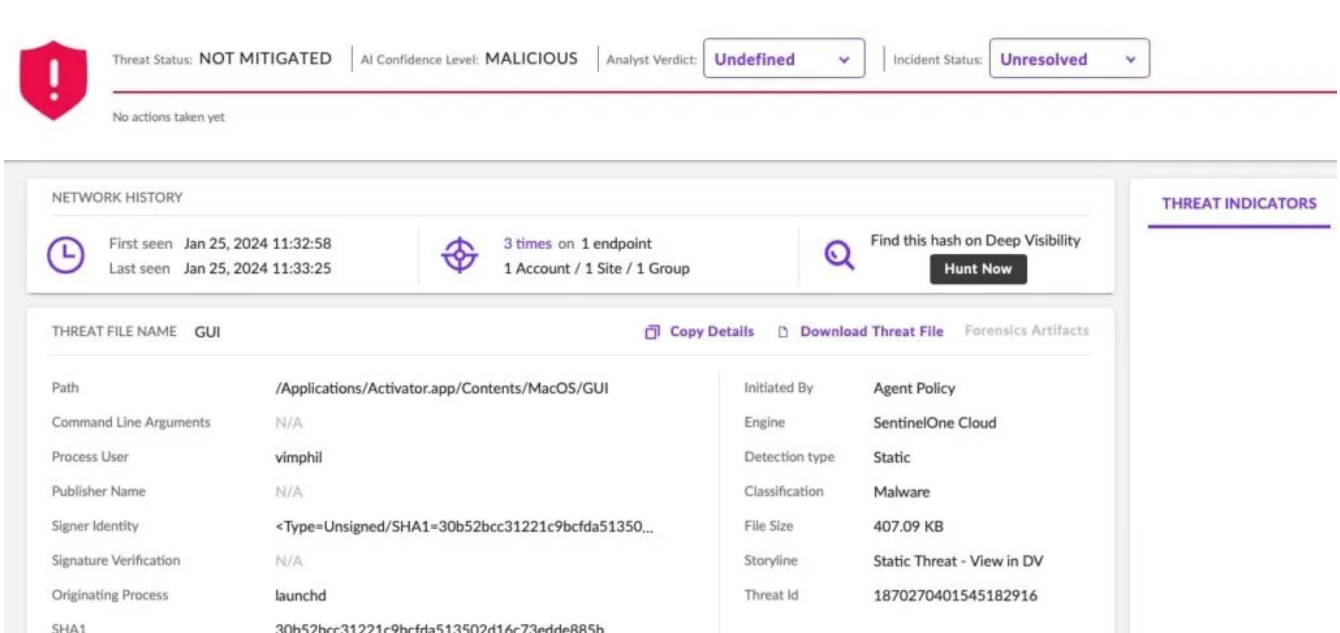
Further Stages

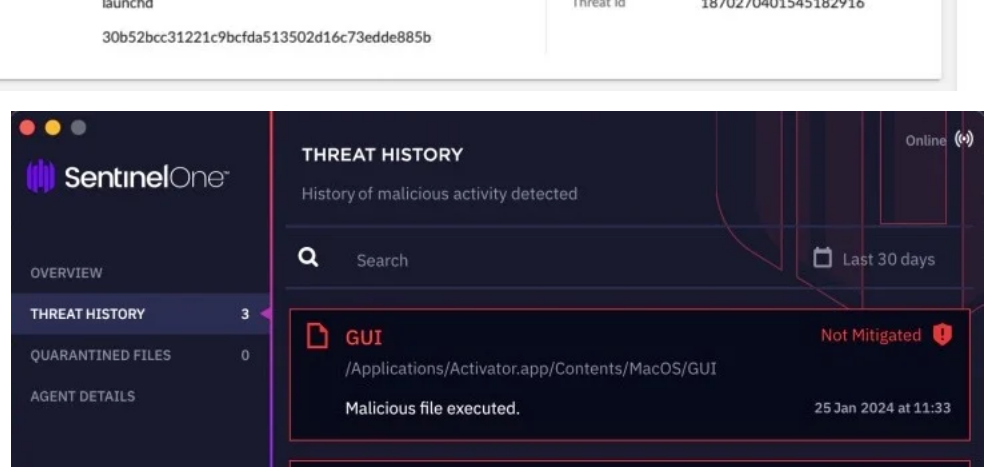
The Activator malware functions as a Stage 1 installer and downloader. The `tool` binary constructs a hardcoded domain name string and, according to Kaspersky researchers, retrieves TXT records for this domain from a DNS server. We were unable to confirm this in our tests, but the previous research suggests that the malware uses a novel technique of retrieving base64-encoded messages from the snippets contained in the DNS responses. These are then decrypted in-memory and were seen to contain a Python script which reached out to a further remote server to download the next stage.

The content of these encrypted messages could change according to the operator's whim, but in the observed case the final stage turned out to be a Python backdoor that allows the operator to execute arbitrary commands on the infected device. More details on this stage can be found here.

SentinelOne Detects macOS.Bkdr.Activator

The campaign is ongoing and we continue to track and identify new malicious samples. When the policy is set to 'Protect', the SentinelOne agent blocks execution of malicious samples. With the policy set to 'Detect Only', an alert is raised and the sample may be allowed to run for the purposes of observation.





Indicators of Compromise

File Paths

/tmp/python-3.9.6-macosx10.9.pkg

/Applications/Activator.app/Contents/MacOS/GUI

/Applications/Activator.app/Contents/Resources/tool

```
[~]/Library/LaunchAgents/launched.%@.plist
#regex:
/Library/LaunchAgents/launched.[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}\.plist
```

SHA1 Mach-Os

01223c67c44b9cb893576c624ceeb6971d7c8a64
02a38a5dd5dcff4354fab26601dd766c1d24293e
03c4a36c06c12e3420bd410a9600e09ddb4b4211
07da6661657d72a4d9fc14990bb57f46514318a9
08503aca7610a83aeb55d5cf68be16b221f677bf
14f6e7759541de4c31e6cdc5efd4059363b748a9
192fd322a6c4df2bb0e3d743dfe84d30c82512bd
1acaf1e08a03137827b9ef1972198cf9b52d0e15
1b434829544a5a63101e4d0e45ddb65ec840c841
21a5895c184b047c7b9aa7aa4f6451acbc8be826
21e6691d8466ecc6fbf25481cc33338ad47caf5c
25e12022e796d77f2496c3c2090febd048015a9f
28de5c653b938626b5c2663de07ec3affb61da7a
29f8c0f7f3a70ec114ac3cef2a47f0c285138fdb
2c6c43cf0655a2ed0d155ea12cfb100f1fc1f770
2c6d7642dd442d1e50985b938a4c5d827720b8b2
2e0159157a2443fe41abc1643d75cc923cda6896
2f26dc03de6ad3e8c7853588a96c524b5093d37e
315b793de51286b03fdedfd7bca1aa8885dfabb8
341e215d527c058d17c82ab34e4fc392a8d20575
343f788d605e9433aebc40edc3d1d621b11aef38
38d38f96558d3a476d9cf0b319299d069ae629e4
392377835b20d2faca7f40c5ea6959f8be0ca586
3a9a511b32753de5e3824abc91a1969bf12fbb47
3bac1bb68a996b0524d1082ec810d6af33061a50
429a81049145a7c03ec39e7d23a20a74d89d6dd9
4f2d4e69abf124edff096870271c4e1942ecef12
55d893acd26927a66583c200377f10baffc06347
5facd492d920ba088acb32d311ede7ae2190c7fd
5fd1f90079bfe29d519ab59380ab9d152e837b6d
61cf0c13d58bb03eaf8886e599132581f96a8585
65ca8d43bc622561d3b9b990873cb82ed2b7db6d
6bc6586134013472c5020e08648c946f5da859aa
719efeae3e91ba89222c8118ad76790cf996ae79
72c2469669b1aa50e0dc356dfc036a405ce26ef3
7966a3cdf552e698c6861849479cb25fb2fe22c7
7ebf2eba7be3535c6afd1195305f683a8d46f45a
8133447d1bfd6a704dbee353cecfa8105bdc324a
8c78b2b159894abf5dfaa08a4cd8b1b79aabe446
8d9f0539f82609de097c244d2c8182f7f240545f
8ecf86ee0eb436e30508b22bcda89585bf5a5613
9089265798cfd830240e1bb981df6e61aea49692
90ffd2f23d0c57c7b3becd52525d31aadcb142ba
92b476221f3b88de74e31aca92c44eb8ae8e1c6c
98e9bb5de5d8f487f84bca9276905a87a76d3bb4
9c75698e5ec05c3613510e866ef37673e1649536
a1bc32090d7a9599d14e5310ffd981727cec4d9a
a2a6948d39a3b1239d0e83792f3178c338aaefb6
a3b9ea16b0d44e835d6458db44c018349f1cff3f
a5a28411bffe4efb72c99a63d234bffdd83bafef
a6fb4aaebd82681b5e5fac086cb4a41c7d64b718
b11d8ba52cef7fc9cd4b224a780bc2440afcfb82
bc51a249ade7b619da3ad4d3593176381f114b01
c4e9f2bc657d32c9e642274c056b3d4a8e0bbb06
c74d70da36badfa1fb4914494d4e952fa56fdbb1
caadd51d6191966002986f5529ab3b60622f9a03
cd4d2e325fd4741bf7c1918e9f341a3bc0e2c45c
d326b6f10d91965282ba0eb0041f2bb3dc0c004b
d58823309eeed0a40287d1df22ce799a672483db
d5b4ba66b24becfce2944a0df7b5d36f2a617ebf
d73cb24b88bdeb29ea09a867d67006061f3d9464
db49f7b2ebb06eba1a821ed9a0050ca36a38d31e
dc64a04830d9209142c72937cd348d581afbad09
dcb8efd9817a46f79021afcad9ea67ef4c898ff6
def1ca81e74dad6bef7cd37d896d9521afd3e19e
e18c9dff96ba0b982cbfd1911db24f974db82cce
e439e6a35fe685b909e8656fed03b4c2ae8533cd
e591b784a7a6783580e8674ff1b263d5a6d91e86
e85cc29f9ea7c7cfcb31450cecaed85bc0201d32
e8613f03b1cbebb6c6fa42a65aef59ab547a8a59
eca71e86d45b43a558f1f05acd6fdbf48c79f097
ee90f40748c4bd0ba78abbf113a6251f39a5bbd5
f3f498574f91da8fc4a69e5ae35dbfcb058abb7b
fa08c5f4c6dbb5f32288ea05ed558ffcd273f181

Like this article? Follow us on LinkedIn, Twitter, YouTube or Facebook to see the content we post.

Read more about Cyber Security

- SmoothOperator | Ongoing Campaign Trojanizes 3CXDesktopApp in Supply Chain Attack
- BlueNoroff | How DPRK's macOS RustBucket Seeks to Evade Analysis and Detection
- The Many Faces of Undetected macOS InfoStealers | KeySteal, Atomic & CherryPie Continue to Adapt
- Geacon Brings Cobalt Strike Capabilities to macOS Threat Actors
- Cloud Credentials Phishing | Malicious Google Ads Target AWS Logins
- Enterprise Security Essentials | Top 12 Most Routinely Exploited Vulnerabilities