

Malware Analysis Spotlight: OSAMiner Uses Run-Only AppleScripts to Evade Detection - VMRay

By VMRay Labs

Published: 2021-01-14 · Archived: 2026-04-05 14:30:16 UTC

This week the team at [SentinelLabs released an in-depth analysis of macOS.OSAMiner, a Monero](#) mining trojan infecting macOS users since 2015. The authors of macOS.OSAMiner used run-only AppleScripts which made attempts at further analysis more difficult.

In 2020, the SentinelLabs Team discovered that the malware authors were evolving their evasion techniques, adding more complexity by embedding one run-only AppleScript inside another. We analyzed one of the latest samples “[com.apple.4V.plist](#)” using [VMRay Analyzer](#). In this [Malware Analysis Spotlight](#), we will showcase the key behaviors identified during the dynamic analysis.

Note, at the time of analysis this sample of OSAMiner had a 2/60 detection rate on VirusTotal.

OSAMiner Analysis

The “com.apple.4V.plist” file is placed in ~/Library/LaunchAgents by the original dropper and disguised as a Property list configuration file (PLIST) while it is in fact a compiled AppleScript.

Straight away, we see that a number of VMRay Threat Identifier (VTI) rules hit and the sample is classified as malicious. From the Overview Tab, we can see the main behaviors of the sample including network connectivity, file dropping behavior, and system information gathering. Now we can dig deeper into each of these characteristics.

MALICIOUS

Classifications
PUA

Threat Names
Mal/HTMLGen-A
Gen:Variant.Application.MAC.Miner.2

DYNAMIC ANALYSIS REPORT

Created 7 hours ago

com.apple.4V.plist.script

Apple Script

Overview

Network

Behavior

Files

AV & YARA

IOCs

Environment

VMRay Threat Identifiers (12 rules, 40 matches)

Score	Category	Operation	Count	Classification
4/5	Reputation	Contacts known malicious URL	1	-
3/5	Anti Analysis	Creates an unusually large number of processes	1	-
2/5	Discovery	Reads network adapter information	1	-
2/5	Network Connection	Performs DNS request	3	-
2/5	Execution	Drops binary file	3	-
2/5	Execution	Executes dropped binary file	1	-
2/5	Discovery	Enumerates running processes	1	-
2/5	Antivirus	Suspicious content was detected by heuristic scan	3	-
2/5	Reputation	Known suspicious file	2	PUA
1/5	Discovery	Reads system data	21	-
1/5	Network Connection	Connects to remote host	3	-

The Network Tab shows multiple C2 connections. The first request to budaybu100001[.]com:8080 returns the second-stage URL embedded in the string “-=-=-=” as a marker. Interestingly, there are two URLs that were returned. The second one might be a fallback or used by another variant of the family.

Overview

Network

Behavior

Files

AV & YARA

IOCs

Environment

General


- 0.44 KB total sent
- 1.56 MB total received
- 2 ports: 80, 8080
- 2 contacted IP addresses
- 3 URLs extracted
- 2 files downloaded
- 0 malicious hosts detected

DNS

- 3 DNS requests for 2 domains
- 1 nameserver contacted
- 0 total requests returned errors

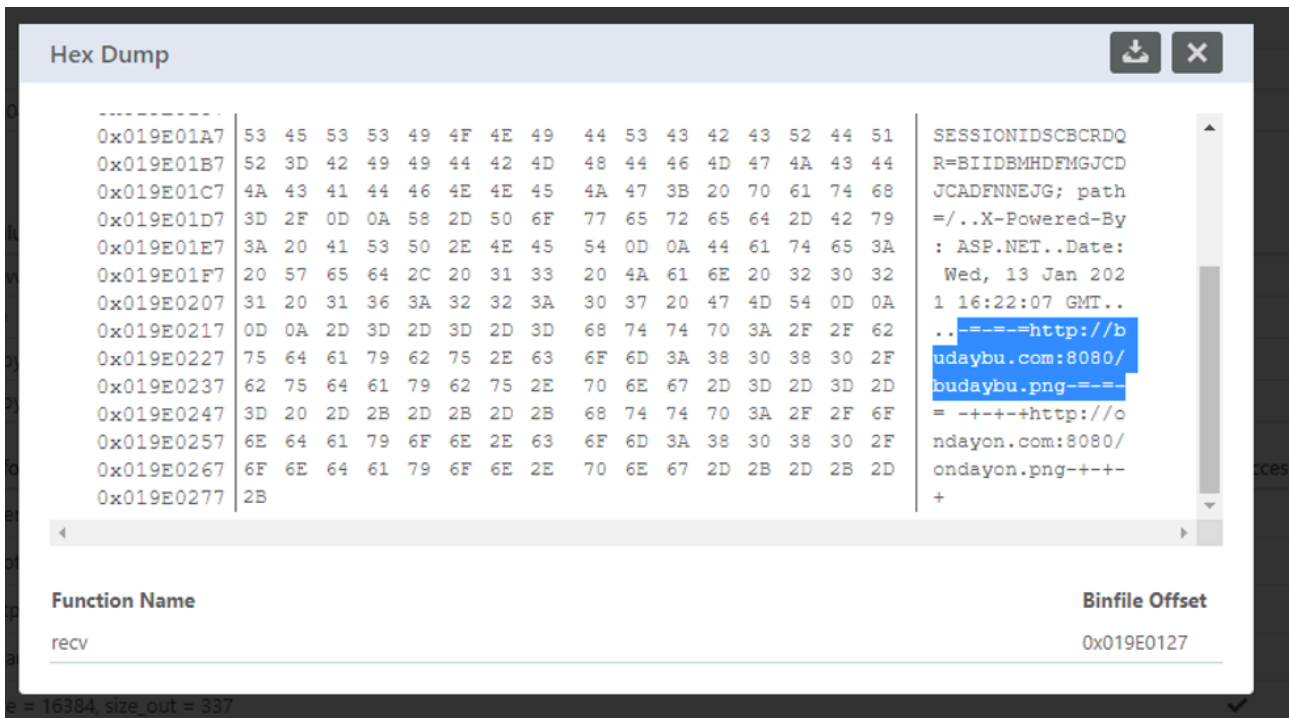
HTTP/S

- 3 URLs contacted, 1 servers
- 1 sessions, 0.00 KB sent, 0.00 KB received



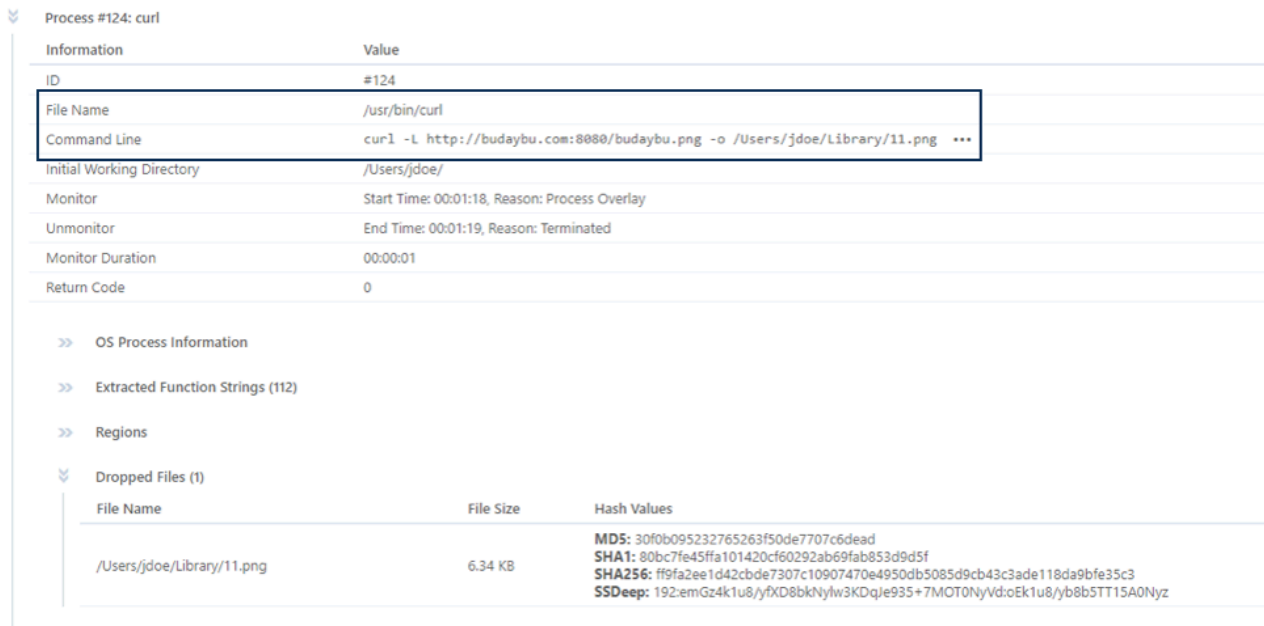
1 Host

Requests	Severity	Method	URL	Response	Dest. IP	Dest. Port	Verdict
www.budaybu100001.com 80, 53, 8080	-	GET	http://www.budaybu100001.com:8080	-	43.249.204.231	80	CLEAN
		GET	http://budaybu.com:8080/budaybu.png	-	43.249.204.231	80	CLEAN
		GET	http://budaybu.com:8080/ssl.zip	-	43.249.204.231	80	MALICIOUS



The second stage is another compiled AppleScript stored at ~/Library/11.png. All downloads are performed using curl which is clearly visible in the Behavior Tab. The second stage is again executed using “osascript” and has two main tasks:

1. Download and extract the third stage mining payload
2. Write the mining configuration (pools.txt, config.txt, cpu.txt)



Process #128: osascript

Information	Value
ID	#128
File Name	/usr/bin/osascript
Command Line	osascript /Users/jdoe/Library/11.png ***
Initial Working Directory	/Users/jdoe/
Monitor	Start Time: 00:01:19, Reason: Process Overlay
Unmonitor	End Time: 00:02:54, Reason: Terminated by Timeout
Monitor Duration	00:01:34
Return Code	Unknown

>> OS Process Information

>> Extracted Function Strings (27)

>> Regions

↳ Dropped Files (2)

File Name	File Size	Hash Values
pools.txt	228 bytes	MDS: 654c34267d9a2ad88dd410d49b39eb SHA1: 00a4790f91bf6754a913414f9d304061100f42 SHA256: f1ac4265b7ad09f11818b50bec3161a15f88940ed676163c0dd6f4e6df267907 SSDeep: 6R+IHQ/1nktH4mWth/DFIXDfcdhtPshfDgVkoQXRiw/0AfvXD8xHPWLD
config.txt	343 bytes	MDS: 29176077682e1a92e8cee52118043479 SHA1: 40247e968e0f29062ce2e7a24e642648e4ea9e09 SHA256: 2fcaa943f404a12c8b951045b02d5ac58ef8222196518ea6624bddd178ba978c SSDeep: 6.MdfSpz6ODJf9M2KYcWKMhujKMHK0oicK8Xrj73wivqVBAUvVkcldHh:MdqoJ2KGxJKmqBoKdti73NSmZdHh

↳ Modified Files

File Name	File Size	Hash Values
/users/jdoe/library/caches/com.apple.90/cpu.txt	117 bytes	MDS: 657714d2aca5f0bcc4e7f4f6dbc32c93 SHA1: 6eafaf73e8fb121ed1ba37bf604f9d10fad827 SHA256: ac8eb3cce3b254d7b1a37d98a4a1bbe7f3de981bb3cb8e51775cd7355aa31865 SSDeep: 3z26cpl+fkBk62HhQhEQ+0DOQkFkMz643fkzgLweENOIQkqM

The third stage is a zip file containing two dynamic libraries (dylibs) and finally a Mach-O binary, again disguised as a PLIST which can be clearly seen in the Files Tab.

/Users/jdoe/library/Caches/com.apple.90/ssl.zip Downloaded File ZIP

MIME Type	application/zip
File Size	1.55 MB
MD5	6bd01c8ddbcca5ed4b5dfe62518865f1
SHA1	40012f5c6a90a2b135fa552f975b9be95bfab53a
SHA256	d63e48873eec22c1be1823209c4c9ff7efe25ebe6607bfac8526dbba511704d8
SSDeep	49152:69d2LPU66yFwqX/qXcSjC0zy1H2WALubSzlxcc:6hqfWqXSsSFEWrbSzll
ImpHash	-

↳ Archive Information

Number of Files	3
Number of Folders	1
Size of Packed Archive Contents	1.55 MB
Size of Unpacked Archive Contents	3.43 MB
File Format	zip

↳ Contents (3)

File Name	Packed Size	Unpacked Size	Compression	Is Encrypted	Modify Time	Severity
openssl/lib/libssl.1.0.0.dylib	213.61 KB	517.68 KB	Deflate	✗	2020-01-20 16:42 (UTC+1)	CLEAN
openssl/lib/libcrypto.1.0.0.dylib	1.09 MB	2.26 MB	Deflate	✗	2020-01-20 16:40 (UTC+1)	CLEAN
ssl4.plist	256.01 KB	682.20 KB	Deflate	✗	2020-01-20 16:37 (UTC+1)	SUSPICIOUS

The screenshot displays a file analysis interface. At the top, the file path is `/Users/jdoe/library/caches/com.apple.90/bc.t_s1jsds`, labeled as a 'Dropped File' and 'Binary'. Below this, a table lists various metadata fields:

Also Known As	/Users/jdoe/library/caches/com.apple.90/ssl4.plist (Dropped File) ssl4.plist (Embedded File)
Parent File	/Users/jdoe/library/Caches/com.apple.90/ssl.zip
MIME Type	application/x-mach-binary
File Size	682.20 KB
MD5	deb6c97315615faa44a0ac07244e7570
SHA1	cfb1a0cd345bb2cbd65ed1e6602140829382a9b4
SHA256	97febb1aa15ad7b1c321f056f7164526eb698297e0fea0c23bd127498ba3e9bb
SSDeep	6144:hrrmr3fQhN/WXgC5gP0Er5MhsiFoTvcfnDPNlb/V53bzQfC0RzcZui2SEX9QcPQwrVsmr3EsX+0EreFGmB/U3EccSEtL6hM
ImpHash	-

Below the metadata, the 'AV Matches (1)' section shows a single match:

Threat Name	Verdict
Gen:Variant.Application.MAC.Miner.2	SUSPICIOUS

The 'Mach-O Information' section provides details about the binary's architecture and flags:

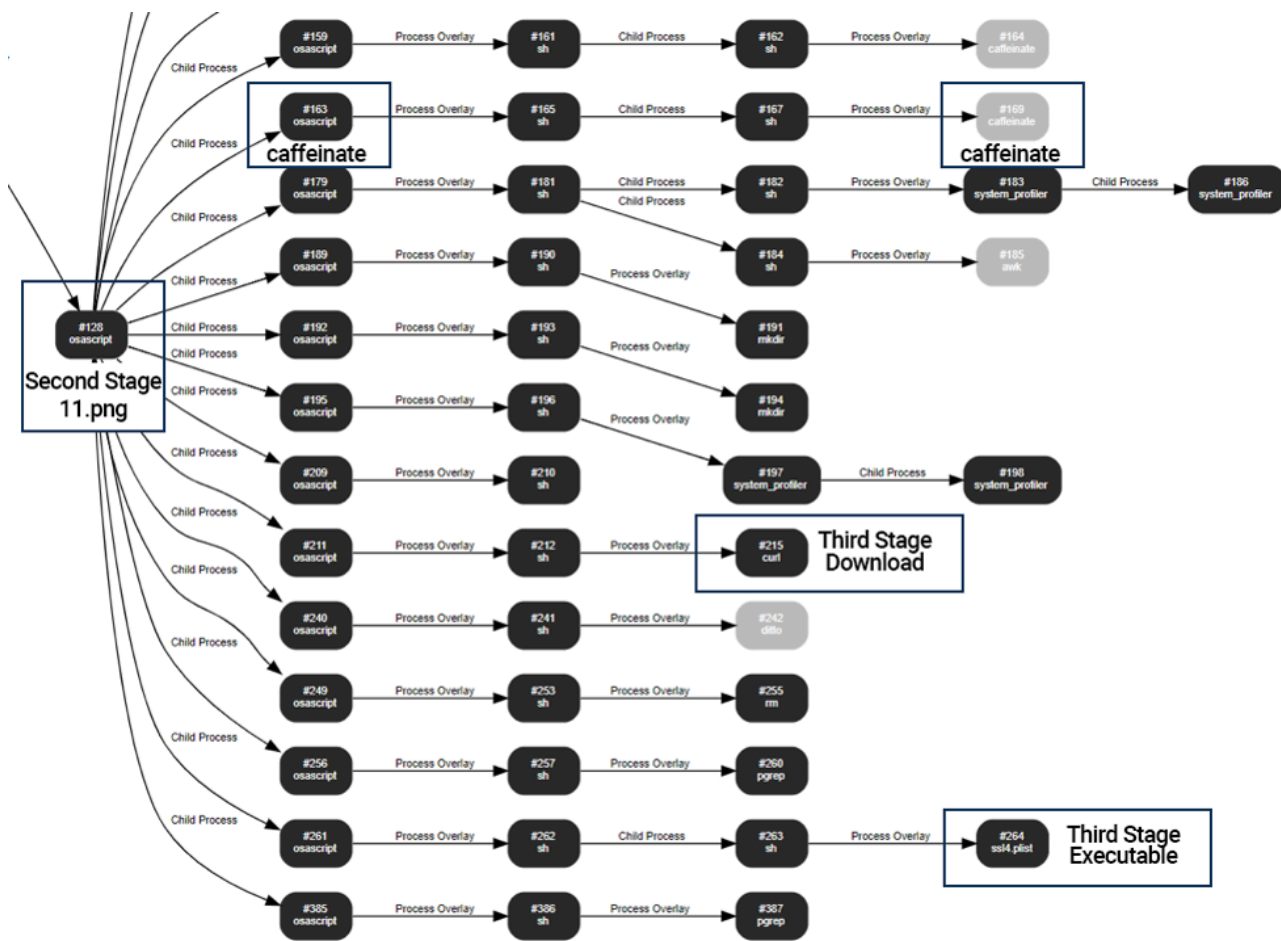
Arch Type	CPU_TYPE_X86_64
Arch Subtype	CPU_SUBTYPE_X86_64_ALL
Type	Executable
Flags	MH_NOUNDEFS, MH_DYLDLINK, MH_TWOLEVEL, MH_WEAK_DEFINES, MH_BINDS_TO_WEAK, MH_PIE
UUID	dfa5291e-15fd-3bfd-b293-12978c99bdfe
Entry Point	0x100007b80

At the bottom, there are expandable sections for 'Segments (4)', 'Imported Libraries (4)', and 'Load Commands (10)'.

In addition, the second stage uses the system tool “caffeinate” to prevent the machine from going to sleep while the first stage will continuously query the running processes for common AV programs using the ps command:

```
sh -c ps ax | grep -E '360|Keeper|MacMgr|Lemon|Malware|Avast|Avira|CleanMyMac' | grep -v grep | awk '{print $1}'
```

All of these actions are performed using sub-processes so they can be observed in the process graph and process overview.



#262	0x24d	Process Overlay	sh	sh -c cd ~/library/Caches/com.apple.90; ~/library/Caches/com.apple.90/ssl4.plist && /dev/null 1 & exit;
#263	0x24e	Child Process	sh	
#264	0x24e	Process Overlay	ssl4.plist	/Users/jdoe/library/Caches/com.apple.90/ssl4.plist
#265	0x24f	Child Process	osascript	
#266	0x24f	Process Overlay	sh	sh -c ps ax grep -E '360 Keeper MacIgr Lemon Malware Avast Avira CleanlyMac' grep -v gre p awk '{print \$1}'

As we can see, this sample uses a different kind of evasion, using a rather uncommon file type, a compiled AppleScript, disguised as a PLIST file. This file type won't have a problem running on a victim's machine but it is difficult for security teams to analyze because of the inherent [obfuscation](#) and limited tooling available.

Running the sample in VMRay gives analysts an immediate view into the key behaviors, characteristics, and IOCs. Within 2 minutes of analysis time, analysts can see a majority of the sample's behavior, compared to hours of manual reverse engineering. And for deeper analysis, the second and third stages are visible and available from the VMRay Analyzer Report.

IOCs

Sample

com.apple.4V.plist

df550039acad9e637c7c3ec2a629abf8b3f35faca18e58d447f490cf23f114e8

Second Stage

~/Library/11.png

ff9fa2ee1d42cbde7307c10907470e4950db5085d9cb43c3ade118da9bfe35c3

Third Stage

~/Library/Caches/com.apple.l0/ssl4.plist

97febb1aa15ad7b1c321f056f7164526eb698297e0fea0c23bd127498ba3e9bb

AV Detection Script embedded in First Stage

~/Library/k.plist

0cc04703ae218b0217e1b025de60cec82087e0774eb59b984419949cee5c2173

Contacted URLs

hxxp://www.budaybu100001[.]com:8080

hxxp://budaybu[.]com:8080/budaybu.png

hxxp://ondayon[.]com:8080/ondayon.png (possibly backup URL)

hxxp://budaybu[.]com:8080/ssl.zip

budaybu[.]com:8888 (mining pool address)

List of Queried Processes

360

Keeper

MacMgr

Lemon

Malware

Avast

Avira

CleanMyMac

Source: <https://www.vmrays.com/cyber-security-blog/osaminer-uses-applescripts-evade-detection-malware-analysis-spotlight/>