FINDING WALDO IN THE APPLE UNIFIED LOG

OBJECTIVE BY THE SEA 3.0

JAI MUSUNURI | ERIK MARTIN

AGENDA

- Why Are We Here?
- Introduction to the Unified Log
- Acquiring the Unified Log
- Parsing the Unified Log
- Finding Waldo
- Conclusion

JAI MUSUNURI

Principal Consultant, CrowdStrike Services

- CrowdStrike Services macOS forensics/IR lead
- Certified Blacklight Examiner
- AutoMacTC developer

ERIK MARTIN

Associate Consultant, CrowdStrike Services

- CrowdStrike Services macOS forensics/IR analyst
- Certified Blacklight Examiner
- AutoMacTC developer

CROWDSTRIKE

WHY ARE WE HERE?

INCIDENT RESPONSE (IR) REQUIRES THE RIGHT STUFF

- Incident responders are sent in to put out “fires” during security incidents
- Firefighters need the right tools to put fires out quickly:
  - Fire truck
  - Hoses
  - Protective gear
  - Firefighting knowledge
- Incident responders need the same:
  - Forensic tools
  - Forensic artifacts
  - Forensic analysis skills

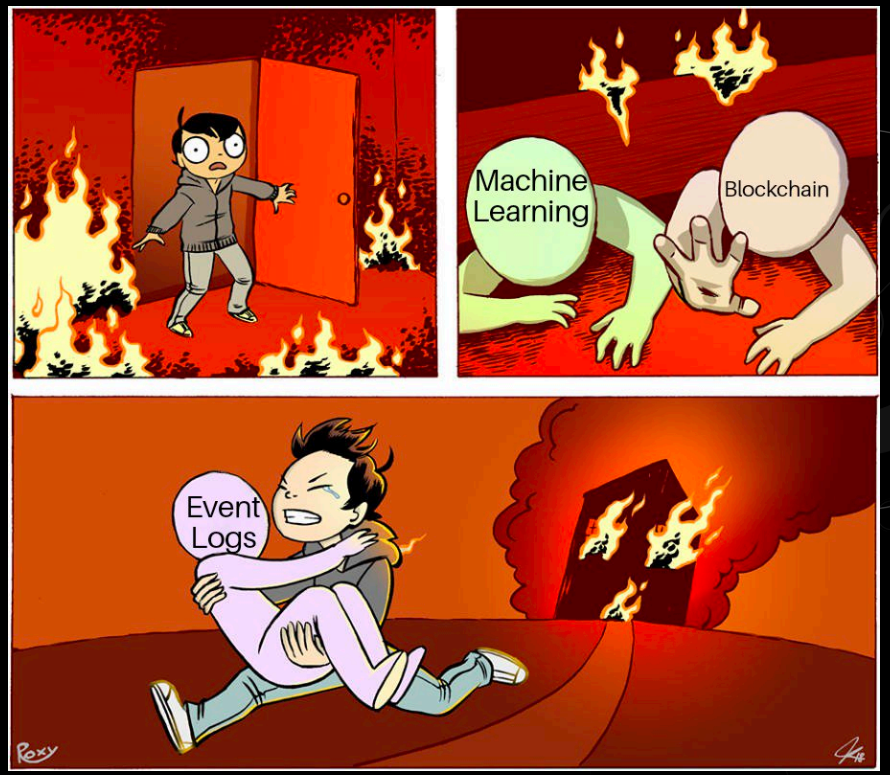
WHAT ARE WE DISCUSSING HERE?

-> How can we leverage the Apple Unified Log to improve our incident response efforts? <-

We support this with an understanding of the unified log's:

- Internals
- Acquisition
- Parsing

WHY DO WE CARE ABOUT LOGS DURING IR?



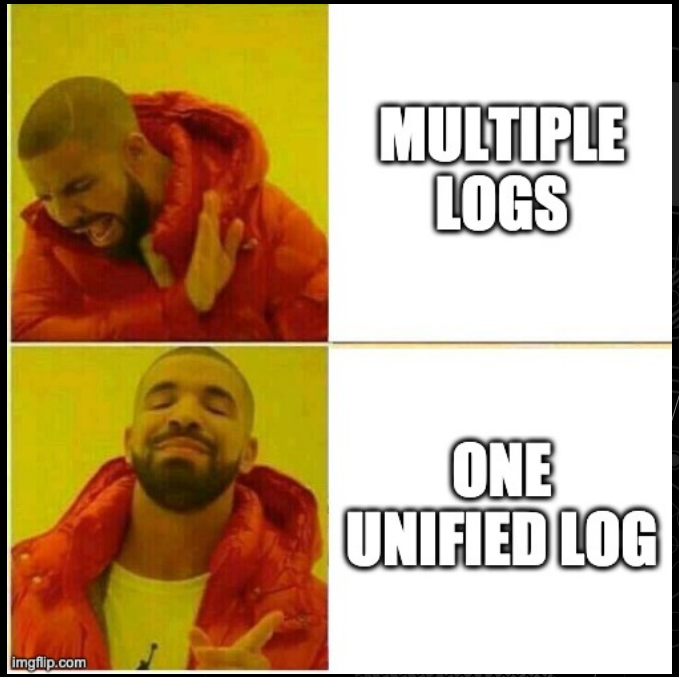
CROWDSTRIKE

INTRODUCTION TO THE UNIFIED LOG

PURPOSE

Apple announced the Unified Log at WWDC ‘16

- Used to assist with debugging
- Single logging mechanism
- Designed to replace traditional Unix logging
- Form of standard Logging mechanism across iOS, macOS, tvOS, and watchOS
- Maximum amount of data in as wide a timeframe as possible



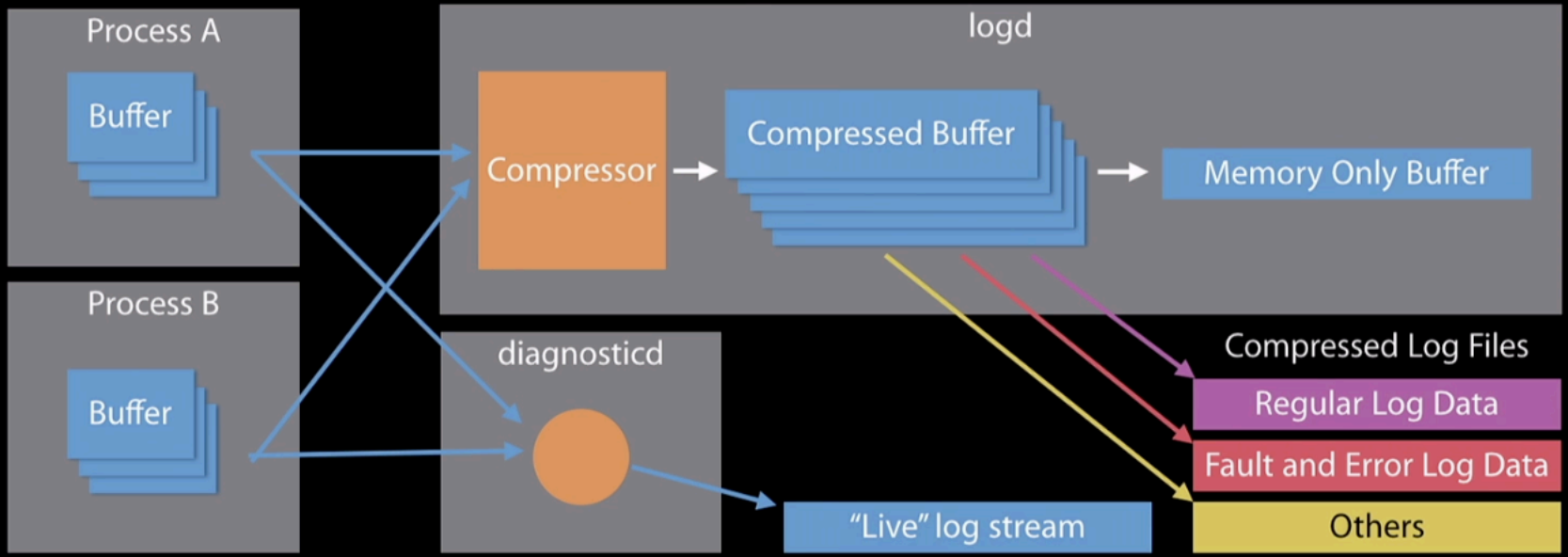
WHERE AND WHAT THE UNIFIED LOG LOGS

- Components found in two locations
  - Var/db/diagnostics - .tracev3 files
  - /var/db/uuidtext. - support files
- Logging is centered out the subsystem and category combo
  - Subsystem: com.apple.objectivebythesea;
  - Category: Conference_Prod, Conference_Test
- Each Log has a level determined by the API used
  - Basic Levels – Default, Info, and Debug
  - Special Levels – Error and Fault

Message Level	Enabled	Destination
Default	Always	Disk
Info	Yes	Memory
Debug	No	N/A
Error	Always	Disk
Fault	Always	Disk

https://developer.apple.com/videos/play/wwdc2016/721/

UNIFIED LOG ARCHITECTURE - HOW IT WORKS



https://developer.apple.com/videos/play/wwdc2016/721/

UNIFIED LOG STRUCTURE

- Log Fields
  - Useful data collected and parsed
- Common Log Fields
  - Data & Time
  - Process
  - Message
- Signposts

	10.12 Sierra	10.13 High Sierra	10.14 Mojave	10.15 Catalina
Log Fields	16	22	27	27
Signposts	N/A	4	5	5

	Predicate	Event Type	Log Type	Signpost Scopes / Types
10.14 Mojave	16 different fields	22 different fields	27 different fields	Text
10.15 Catalina	N/A	4	Text	Text

https://eclecticlight.co/2019/10/16/inside-catalinas-unified-log-how-has-it-changed/

UNIFIED LOG EXAMPLE

Date & Time	Process	Message
2019-11-04 22:00:11.114256	screensharingd	Authentication: SUCCEEDED :: User Name: N/A :: Viewer Address: 192.168.107.153 :: Type: Guest Request for Control
2019-11-04 22:00:11.115624	screensharingd	MessageTracer: Falling back to default whitelist
2019-11-04 22:00:11.116488	screensharingd	pref set for session select
2019-11-04 22:00:11.116491	screensharingd	send session select info to viewer
2019-11-04 22:00:11.116491	screensharingd	viewerInitializationFlags 0xc1
2019-11-04 22:00:11.116512	screensharingd	SSAgentOnConsole 11267
2019-11-04 22:00:11.116518	screensharingd	SSAgentOnConsole 11267
2019-11-04 22:00:11.119163	screensharingd	logged in flag 1
2019-11-04 22:00:11.119173	screensharingd	userLoggedInFlag 1
2019-11-04 22:00:11.119175	screensharingd	should start on console pref 0 serialNumflag 0 askFlag 1 userLoggedInFlag 1
2019-11-04 22:00:11.119209	screensharingd	uid 0 wantConsole 1 createLoginWindow 0 waitflag 1 maxtime 45
2019-11-04 22:00:11.119212	screensharingd	LockScreenIsActive check
2019-11-04 22:00:11.119378	screensharingd	CheckLockScreenPIDFile: return 0

Showing: All Messages

screensharingd

Subsystem: Category: Hide

Activity ID: 0 Thread ID: 0x4afa PID: 922

```
Authentication: SUCCEEDED :: User Name: N/A :: Viewer Address: 192.168.107.153 :: Type: Guest Request for Control
```

LOG VOLUME

Unified Log

- 28-30 days of retention
- 30-50 million records
- Logarchive size – 400-800 MB
- Plaintext output – 2-9 GB

Apple System Log (ASL)

- 40-60 MB of data
- 200K – 500K records
- Full data set retained 7 days
- Limited data retained 1 year

System.log

- 30-60 MB of data
- All data retained 7-14 days
- 200K – 400K records

CROWDSTRIKE

ACQUIRING THE UNIFIED LOG

FROM A MACOS SYSTEM

From Disk

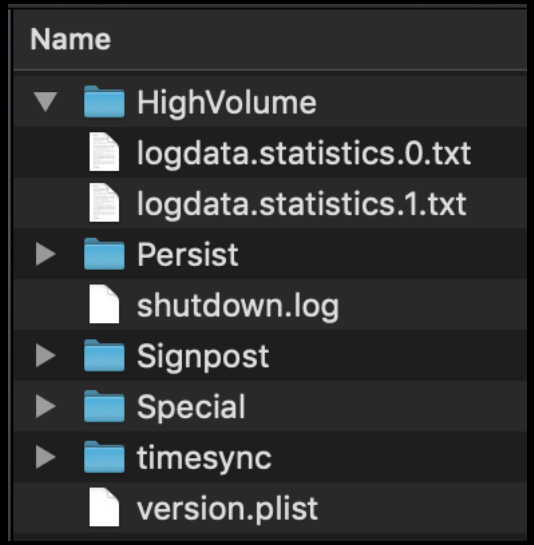
- Reconstruct the unified log from /private/var/db/diagnostics /private/var/db/uuidtext
- Place constituent files from both locations into one directory
- Add .logarchive extension

Live System – Log Command

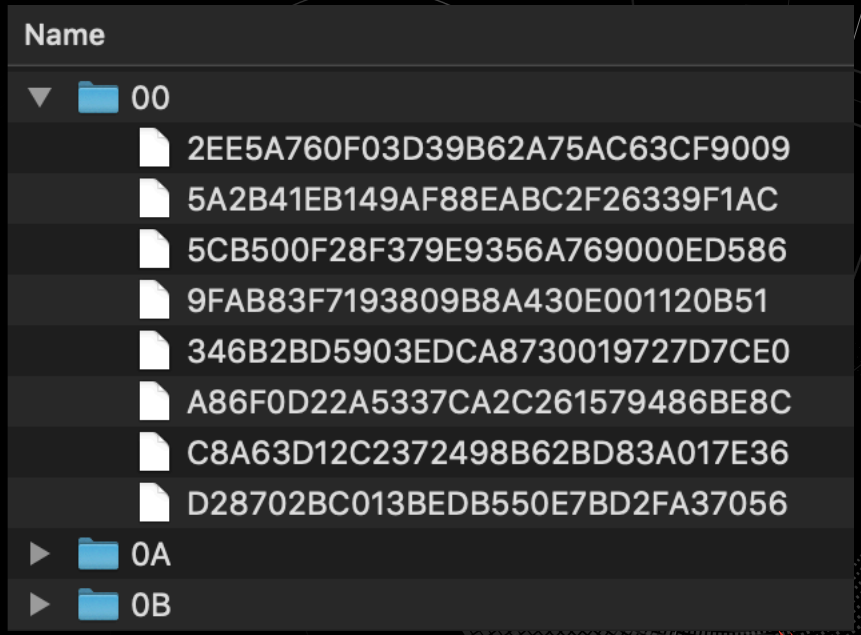
```
usage:
    log <command>

global options:
    -?, --help
    -q, --quiet
    -v, --verbose

commands:
    collect         gather system logs into a log archive
    config          view/change logging system settings
    erase           delete system logging data
    show            view/search system logs
    stream          watch live system logs
    stats           show system logging statistics

further help:
    log help <command>
    log help predicates
```

FROM DISK

/private/var/db/diagnostics



/private/var/db/uuidtext



https://www.blackbagtech.com/blog/accessing-unified-logs-image/

LOG COLLECT

```
usage: log collect [<options>]

description:
    Collect the system logs into a .logarchive that can be viewed with
    `log show` or Console.app.

    Requires root.

options:
    --last <num>[m|h|d]          Collect logs starting <num>[m|h|d] ago
    --output <path>              Output log archive to the given path
    --size <num>[k|m]            Limit log collection to the given size
    --start <time>               Collect logs starting at the given time

notes:
    If an output path is not specified, system_logs.logarchive will be created
    in the current directory. If the output path is a directory, a file named
    `system_logs.logarchive` will be created in the specified directory. If the
    path contains the extension .logarchive, a new logarchive will be created
    with that name at the specified path.

valid time formats:
    'Y-M-D H:m:s+zzzz', 'Y-M-D H:m:s', 'Y-M-D', '@unixtime'

examples:
    log collect --output ~/mylogs.logarchive
    log collect --output /tmp
    log collect --start "2016-04-12" --output /Users/test --size 20m
    log collect --start "2016-04-12 06:30:00"
```

LIVE SYSTEM – LOG SHOW

```
description:
    Show the contents of the system log datastore or a log archive.
    Output contains only default level messages unless --info and/or
    --debug are specified.

options:
    --[no-]backtrace              Control whether backtraces are shown
    --[no-]debug                  Control whether "Debug" events are shown
    --[no-]info                   Control whether "Info" events are shown
    --[no-]loss                   Control whether message loss events are shown
    --[no-]signpost               Control whether signposts are shown
    --color <mode>                Control color output (valid: auto, always, none)
    --end <date>                  Display events up to the given end date
    --last <num>[m|h|d]           Display recent events up to the given limit
    --predicate <predicate>       Filter events using the given predicate
    --source                      Annotate output with source file and line-number
    --start <date>                Display events from the given start date
    --style <style>               Output format (valid: syslog, json, compact)
    --timezone local | <tz>       Use the given timezone when displaying event timestamps
    --mach-continuous-time        Print mach continuous time timestamps rather than walltime

valid time formats:
    'Y-M-D H:m:s+zzzz', 'Y-M-D H:m:s', 'Y-M-D', '@unixtime'

predicate usage:
    Filter predicates follow the NSPredicate format described at:
    https://developer.apple.com/library/content/documentation/Cocoa/Conceptual/Predicates/AdditionalChapters/Introduction.html

    For predicate field/type details, see `log help predicates`.
```

LIVE SYSTEM – LOG STREAM

```
usage: log stream [options]
   or: log stream [options] --process <pid> | <process>

description:
    Stream events from the system or a given process.

options:
    --color <mode>                  Control color output (valid: auto, always, none)
    --level default | info | debug  Include events at, and below, the given level
    --predicate <predicate>         Filter events using the given predicate
    --process <pid> | <process>     Stream events from the specified process
    --source                        Annotate output with source file and line-number
    --style <style>                 Output format (valid: syslog, json, compact)
    --timeout <num>[m|h|d]          Terminate streaming after timeout has elapsed
    --type activity | log | trace   Limit streaming to a given event type
    --mach-continuous-time          Print mach continuous time timestamps rather than walltime

predicate usage:
    Filter predicates follow the NSPredicate format described at:
    https://developer.apple.com/library/content/documentation/Cocoa/Conceptual/Predicates/AdditionalChapters/Introduction.html

    For predicate field/type details, see `log help predicates`.
```

LIVE SYSTEM – LOG

DEMO

CROWDSTRIKE

PARSING THE UNIFIED LOG

UNIFIED LOG PARSING METHODS

Yogesh Khatri's UnifiedLogReader

- Python-based
- Open-source
- Filtering capabilities
- Works on multiple platforms

Blackbag Blacklight

- Commercial GUI-based application
- Easy to integrate with disk analysis workflow
- Filtering capabilities
- Log export capabilities

Native – log show

- Built in to macOS
- Numerous filtering features, including predicates

COMPARING PARSING OUTPUTS

Test condition: Sample VM image running 10.14.6, logarchive constructed from disk

UnifiedLogReader and Blacklight

- 936,446 records

Log show --info --signpost

- 936,452 records
- Difference reflects 6 timesync records
  - System boot, system clock adjustment

A ‘LOG SHOW’ (AND CONSOLE) BUG

Test condition: Sample VM image running 10.14.6, analysis machines running 10.14.6 and 10.15.3

```
2019-11-06 09:35:00.598742+000( 0xde0b     Default     0x0                   738    0    analyticsd: (CrashReporterSupport)
Saved core_analytics report for ??? version ??? to Analytics_2020-03-01-153510_Johns-Mac.core_analytics
```

log show --info --debug --signpost --timezone UTC <logarchive_from_forensic_image>

```
2020-03-01 23:35:10.383844 0xde0b     Default     0x0                   738    0    analyticsd (CrashReporterSupport) Saved
core_analytics report for ??? version ??? to Analytics_2020-03-01-153510_Johns-Mac.core_analytics
```

UnifiedLogReader

CROWDSTRIKE

FINDING WALDO

“OLD FAITHFUL” APPROACH

Grep, Grep, and more Grep

- Using the log collect command, standard out the file to a text file
- Begin keyword searching

However, some issues with this

- Slower
- Storage location
- Correct syntax

FILTERING

Time based Filtering

- Use --start and --end to pull slices from log SHOW only
- Use --start and --last to pull slices from log COLLECT only

Size based Data Reduction

- Use --size to pull a specific size of logs from log COLLECT only

Log Collect

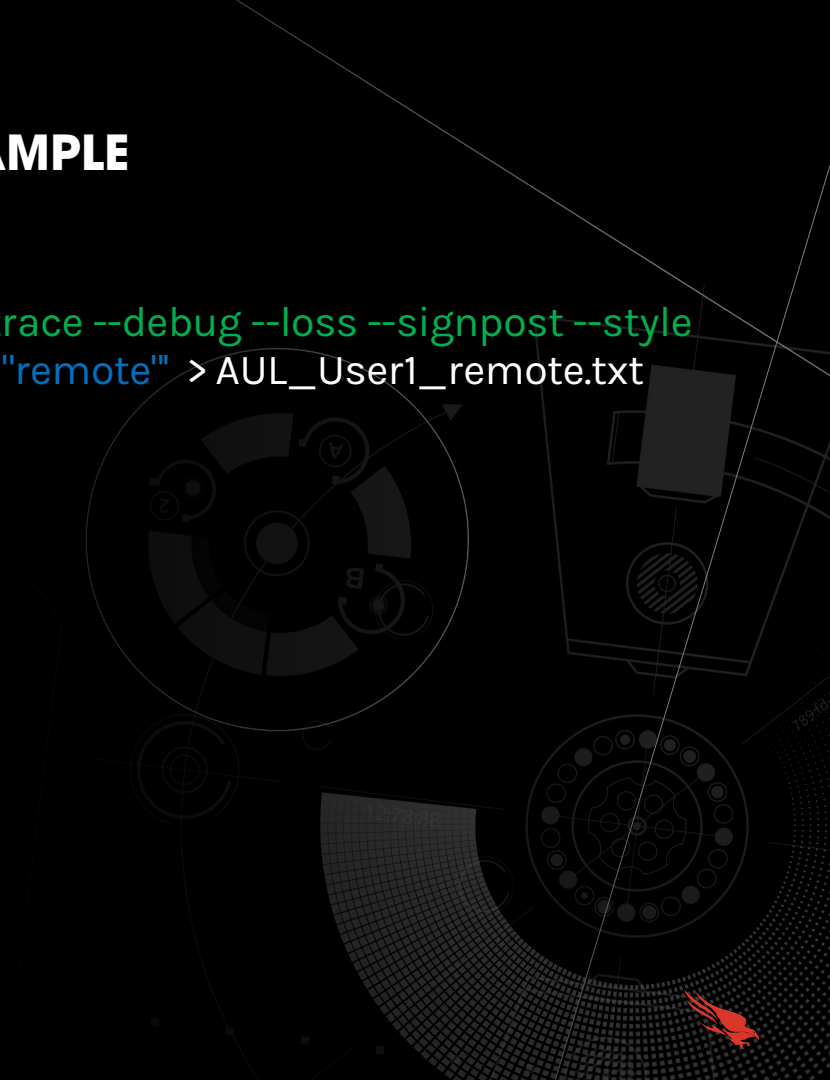
```
options:
    --last <num>[m|h|d]
    --output <path>
    --size <num>[k|m]
    --start <time>
```

Log Show

```
options:
    --[no-]backtrace
    --[no-]debug
    --[no-]info
    --[no-]loss
    --[no-]signpost
    --color <mode>
    --end <date>
    --last <num>[m|h|d]
    --predicate <predicate>
    --source
    --start <date>
    --style <style>
    --timezone local | <tz>
    --mach-continuous-time
```

COMMAND EXAMPLE

- log show User1_unifiedLof.logarchive --info --backtrace --debug --loss --signpost --style syslog --force --predicate 'eventMessage CONTAINS "remote"' > AUL_User1_remote.txt



PREDICATE EXAMPLES

process == “sudo”

Captures command line activity run with elevated privileges

process == “logind”

Captures user login events

process == "tccd"

Captures events that indicate permissions and access violations

process == "sshd”

Captures successful, failed, and general ssh activity

PREDICATE EXAMPLES CONT.

process == "kextd" && sender == "IOKit"

Captures successful and failed attempts to add kernel extensions

process == "screensharingd" || process == "ScreensharingAgent"

Captures events that indicate successful or failed authentication via screensharing

process == "loginwindow" && sender == "Security"

Capture keychain.db unlock events

process == "securityd" && eventMessage CONTAINS "Session " && subsystem == "com.apple.securityd"

Captures session creation and destruction events

MOCK SCENARIO BACKGROUND

Events took place on November 4th 2019

Users reported that Mac was running slow

An unknown application requested the user type their password

IR RESPONSE USING THE UNIFIED LOG
SSHD

```
...
2019-11-04 22:40:43.963081-0800  localhost sshd[1330]: Connection closed by 192.168.87.132 port 41220 [preauth]
...
2019-11-04 22:41:36.051860-0800  localhost sshd[1336]: Accepted keyboard-interactive/pam for john from 192.168.87.132 port 41702 ssh2
...
2019-11-04 22:46:03.394215-0800  localhost sshd[1339]: Received disconnect from 192.168.87.132 port 41702:11: disconnected by user
2019-11-04 22:46:03.394263-0800  localhost sshd[1339]: Disconnected from user john 192.168.87.132 port 41702
```

```
Connection closed by 192.168.87.132 port 41220 [preauth]

Accepted keyboard-interactive/pam for john from 192.168.87.132 port 41702 ssh2

Received disconnect from 192.168.87.132 port 41702:11: disconnected by user
Disconnected from user john 192.168.87.132 port 41702
```

CROWDSTRIKE

IR RESPONSE USING THE UNIFIED LOG SUDO

```
...
2019-11-04 22:30:04.033407-0800  localhost sudo[1264]:     john : TTY=ttys000 ; PWD=/Users/john ; USER=root ; COMMAND=/usr/bin/whoami
...
2019-11-04 22:30:05.170495-0800  localhost sudo[1269]:     john : TTY=ttys000 ; PWD=/Users/john ; USER=root ; COMMAND=/bin/bash
...
2019-11-04 22:30:25.997339-0800  localhost sudo[1276]:     root : TTY=ttys000 ; PWD=/private/var/root ; USER=root ; COMMAND=/usr/sbin/systemsetup -getremotelogin
...
2019-11-04 22:31:13.948650-0800  localhost sudo[1284]:     root : TTY=ttys000 ; PWD=/private/var/root ; USER=root ; COMMAND=/usr/sbin/systemsetup -setremotelogin on
...
2019-11-04 22:40:20.851139-0800  localhost sudo[1317]:     john : TTY=ttys000 ; PWD=/Users/john ; USER=root ; COMMAND=/usr/bin/whoami
...
2019-11-04 22:40:21.927151-0800  localhost sudo[1322]:     john : TTY=ttys000 ; PWD=/Users/john ; USER=root ; COMMAND=/bin/bash
...
2019-11-05 17:29:16.786030-0800  localhost sudo[1691]:     john : TTY=ttys001 ; PWD=/Users/john/Desktop/AutoMacTc ; USER=root ; COMMAND=/usr/bin/python2.7 automactc.py -m all
...
2019-11-05 18:06:10.398145-0800  localhost sudo[1775]:     john : TTY=ttys000 ; PWD=/Users/john/Desktop ; USER=root ; COMMAND=/usr/bin/log collect
```

```
john : TTY=ttys000 ; PWD=/Users/john ; USER=root ; COMMAND=/usr/bin/whoami

john : TTY=ttys000 ; PWD=/Users/john ; USER=root ; COMMAND=/bin/bash

root : TTY=ttys000 ; PWD=/private/var/root ; USER=root ; COMMAND=/usr/sbin/systemsetup -getremotelogin

root : TTY=ttys000 ; PWD=/private/var/root ; USER=root ; COMMAND=/usr/sbin/systemsetup -setremotelogin on

john : TTY=ttys000 ; PWD=/Users/john ; USER=root ; COMMAND=/usr/bin/whoami

john : TTY=ttys000 ; PWD=/Users/john ; USER=root ; COMMAND=/bin/bash

john : TTY=ttys001 ; PWD=/Users/john/Desktop/AutoMacTc ; USER=root ; COMMAND=/usr/bin/python2.7 automactc.py -m all

john : TTY=ttys000 ; PWD=/Users/john/Desktop ; USER=root ; COMMAND=/usr/bin/log collect
```

IR RESPONSE USING THE UNIFIED LOG SCREENSHARINGD

```
2019-11-04 21:58:43.649731-0800  localhost screensharingd[922]: SSDaemon_Checkin  port = 9491 agentPort 11267 effectiveUID 501 sessionID 257 agent pid 923 onconsole 1 vfb 0 loginwindow 0
...
2019-11-04 22:00:11.114256-0800  localhost screensharingd[922]: Authentication: SUCCEEDED :: User Name: N/A :: Viewer Address: 192.168.107.153 :: Type: Guest Request for Control
...
2019-11-04 22:11:55.703212-0800  localhost screensharingd[922]: reached eof
2019-11-04 22:11:55.703244-0800  localhost screensharingd[922]: closing 0
2019-11-04 22:11:55.703255-0800  localhost screensharingd[922]: going to close 6
2019-11-04 22:11:55.703301-0800  localhost screensharingd[922]: going to log accelleration flag 4
2019-11-04 22:11:55.703501-0800  localhost screensharingd[922]:  session not accelerated 0
2019-11-04 22:11:55.703503-0800  localhost screensharingd[922]: viewer->fileCopyInfo 0x0
2019-11-04 22:11:55.703503-0800  localhost screensharingd[922]: viewer->mode 1
2019-11-04 22:11:55.703508-0800  localhost screensharingd[922]: got screen tracker lock
2019-11-04 22:11:55.703509-0800  localhost screensharingd[922]: check for monitoring 0
2019-11-04 22:11:55.703509-0800  localhost screensharingd[922]: stop monitoring screen changes
2019-11-04 22:11:55.916102-0800  localhost screensharingd[922]: set gViewerConnections index 0 to -1
2019-11-04 22:11:55.916103-0800  localhost screensharingd[922]: gViewerConnections[ descriptorIndex ] -1 address 0x10f279510 descriptorIndex = 0
2019-11-04 22:11:55.917903-0800  localhost screensharingd[922]: MVS_FreeInfo
2019-11-04 22:11:55.921107-0800  localhost screensharingd[922]: reset gMaxViewerConnection to -1
2019-11-04 22:11:55.921110-0800  localhost screensharingd[922]: unlocked mutexes
2019-11-04 22:11:55.922860-0800  localhost screensharingd[922]: remove timer
2019-11-04 22:12:10.930560-0800  localhost screensharingd[922]: No viewers so time to exit
...
2019-11-04 22:47:27.245569-0800  localhost screensharingd[1350]: SSDaemon_Checkin  port = 7427 agentPort 20739 effectiveUID 501 sessionID 257 agent pid 1351 onconsole 1 vfb 0 loginwindow 0
...
2019-11-04 22:47:33.285463-0800  localhost screensharingd[1350]: Authentication: SUCCEEDED :: User Name: john :: Viewer Address: 192.168.87.132 :: Type: DH
...
2019-11-04 23:14:23.225492-0800  localhost screensharingd[1350]: reached eof
2019-11-04 23:14:23.225496-0800  localhost screensharingd[1350]: closing 0
2019-11-04 23:14:23.225508-0800  localhost screensharingd[1350]: going to close 5
2019-11-04 23:14:23.225591-0800  localhost screensharingd[1350]: going to log accelleration flag 4
2019-11-04 23:14:23.225764-0800  localhost screensharingd[1350]:  session not accelerated 0
2019-11-04 23:14:23.225765-0800  localhost screensharingd[1350]: viewer->fileCopyInfo 0x0
2019-11-04 23:14:23.226436-0800  localhost screensharingd[1350]: viewer->mode 0
2019-11-04 23:14:23.226442-0800  localhost screensharingd[1350]: got screen tracker lock
2019-11-04 23:14:23.226443-0800  localhost screensharingd[1350]: check for monitoring 0
2019-11-04 23:14:23.230082-0800  localhost screensharingd[1350]: stop monitoring screen changes
2019-11-04 23:14:23.442991-0800  localhost screensharingd[1350]: set gViewerConnections index 0 to -1
2019-11-04 23:14:23.442993-0800  localhost screensharingd[1350]: gViewerConnections[ descriptorIndex ] -1 address 0x1016b2510 descriptorIndex = 0
2019-11-04 23:14:23.445161-0800  localhost screensharingd[1350]: MVS_FreeInfo
2019-11-04 23:14:23.447256-0800  localhost screensharingd[1350]: reset gMaxViewerConnection to -1
2019-11-04 23:14:23.447258-0800  localhost screensharingd[1350]: unlocked mutexes
2019-11-04 23:14:23.448181-0800  localhost screensharingd[1350]: remove timer
2019-11-04 23:14:38.455660-0800  localhost screensharingd[1350]: No viewers so time to exit
```

CROWDSTRIKE

IR RESPONSE USING THE UNIFIED LOG SCREENSHARINGD

```
Timestamp                       Thread     Type        Activity            PID    TTL
2019-11-04 22:40:43.963081-0800 0x6c8a     Info        0x0                 1330   0    sshd: Connection closed by 192.168.87.132 port 41220 [preauth]
2019-11-04 22:41:36.051860-0800 0x6cdb     Info        0x0                 1336   0    sshd: Accepted keyboard-interactive/pam for john from 192.168.87.132 port 41702 ssh2
2019-11-04 22:46:03.394215-0800 0x6cfc     Info        0x0                 1339   0    sshd: Received disconnect from 192.168.87.132 port 41702:11: disconnected by user
2019-11-04 22:46:03.394263-0800 0x6cfc     Info        0x0                 1339   0    sshd: Disconnected from user john 192.168.87.132 port 41702
2019-11-04 22:47:33.285463-0800 0x6fd3     Default     0x0                 1350   0    screensharingd: Authentication: SUCCEEDED :: User Name: john :: Viewer Address: 192.168.87.132 :: Type: DH
```

```
sshd: Connection closed by 192.168.87.132 port 41220 [preauth]
sshd: Accepted keyboard-interactive/pam for john from 192.168.87.132 port 41702 ssh2
sshd: Received disconnect from 192.168.87.132 port 41702:11: disconnected by user
sshd: Disconnected from user john 192.168.87.132 port 41702
screensharingd: Authentication: SUCCEEDED :: User Name: john :: Viewer Address: 192.168.87.132 :: Type: DH
```

CROWDSTRIKE

IR RESPONSE USING THE UNIFIED LOG TIMELINE

```
2019-11-04 22:40:22.049657-0800  localhost espl[1328]: NSHomeDirectory() /var/root
2019-11-04 22:40:22.100420-0800  localhost rapportd[407]: (CoreUtils) [com.apple.CoreUtils:CUWiFiManager] SysMon: ### WiFiManagerClientCopyDevices failed: NULL
2019-11-04 22:40:24.422023-0800  localhost kernel[0]: (Sandbox) sb_user_approval: kTCCServiceSystemPolicyAllFiles for RTProtectionDaem [39]
2019-11-04 22:40:24.422031-0800  localhost kernel[0]: (Sandbox) sb_user_approval: pid 39 responsible for 39
2019-11-04 22:40:24.422035-0800  localhost kernel[0]: (Sandbox) sb_user_approval: kTCCServiceSystemPolicyAllFiles satisifed from cache for pid 39: not approved
2019-11-04 22:40:25.539763-0800  localhost espl[1328]: result = ifconfig
2019-11-04 22:40:25.540021-0800  localhost espl[1328]: running task ifconfig
2019-11-04 22:40:25.550855-0800  localhost ifconfig[1329]: ioctl(SIOCGIFNAT64PREFIX): 12
2019-11-04 22:40:25.550966-0800  localhost ifconfig[1329]: ioctl(SIOCGIFNAT64PREFIX): 12
2019-11-04 22:40:25.551006-0800  localhost ifconfig[1329]: ioctl(SIOCGIFNAT64PREFIX): 12
2019-11-04 22:40:25.551044-0800  localhost ifconfig[1329]: ioctl(SIOCGIFNAT64PREFIX): 12
2019-11-04 22:40:25.551115-0800  localhost ifconfig[1329]: ioctl(SIOCGIFNAT64PREFIX): 12
2019-11-04 22:40:26.806924-0800  localhost sharingd[463]: (CoreUtils) [com.apple.CoreUtils:CUWiFiManager] SysMon: ### WiFiManagerClientCopyDevices failed: NULL
2019-11-04 22:40:29.458243-0800  localhost kernel[0]: (Sandbox) sb_user_approval: kTCCServiceSystemPolicyAllFiles for RTProtectionDaem [39]
2019-11-04 22:40:29.458380-0800  localhost kernel[0]: (Sandbox) sb_user_approval: pid 39 responsible for 39
2019-11-04 22:40:29.458384-0800  localhost kernel[0]: (Sandbox) sb_user_approval: kTCCServiceSystemPolicyAllFiles satisifed from cache for pid 39: not approved
2019-11-04 22:40:32.420737-0800  localhost rapportd[407]: (CoreUtils) [com.apple.CoreUtils:CUWiFiManager] SysMon: ### WiFiManagerClientCopyDevices failed: NULL
2019-11-04 22:40:32.602315-0800  localhost apsd[72]: [com.apple.apsd:daemon] <private> received courierConnectionStatusDidChange from <private>. isConnected? NO
```


```
]: NSHomeDirectory() /var/root
407]: (CoreUtils) [com.apple.CoreUtils:CUWiFiManager] SysMon: ### WiFiManagerClientCopyDevices failed: NULL
: (Sandbox) sb_user_approval: kTCCServiceSystemPolicyAllFiles for RTProtectionDaem [39]
: (Sandbox) sb_user_approval: pid 39 responsible for 39
: (Sandbox) sb_user_approval: kTCCServiceSystemPolicyAllFiles satisifed from cache for pid 39: not approved
]: result = ifconfig
]: running task ifconfig
1329]: ioctl(SIOCGIFNAT64PREFIX): 12
1329]: ioctl(SIOCGIFNAT64PREFIX): 12
1329]: ioctl(SIOCGIFNAT64PREFIX): 12
1329]: ioctl(SIOCGIFNAT64PREFIX): 12
1329]: ioctl(SIOCGIFNAT64PREFIX): 12
463]: (CoreUtils) [com.apple.CoreUtils:CUWiFiManager] SysMon: ### WiFiManagerClientCopyDevices failed: NULL
: (Sandbox) sb_user_approval: kTCCServiceSystemPolicyAllFiles for RTProtectionDaem [39]
: (Sandbox) sb_user_approval: pid 39 responsible for 39
: (Sandbox) sb_user_approval: kTCCServiceSystemPolicyAllFiles satisifed from cache for pid 39: not approved
407]: (CoreUtils) [com.apple.CoreUtils:CUWiFiManager] SysMon: ### WiFiManagerClientCopyDevices failed: NULL
 [com.apple.apsd:daemon] <private> received courierConnectionStatusDidChange from <private>. isConnected? NO
```

CROWDSTRIKE

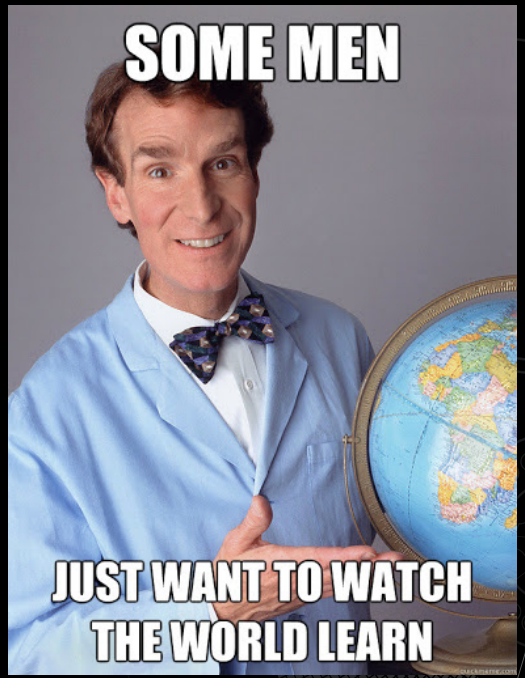
IR RESPONSE USING THE UNIFIED LOG
ESPL

```
2019-11-04 22:25:49.248018-0800  localhost espl[1232]: (libsystem_info.dylib) Created Activity ID: 0x6e20, Description: Retrieve User by
ID
2019-11-04 22:25:49.272560-0800  localhost espl[1232]: NSHomeDirectory() /Users/john
2019-11-04 22:26:55.395294-0800  localhost espl[1232]: result = ifconfig
2019-11-04 22:26:55.395700-0800  localhost espl[1232]: running task ifconfig
...
2019-11-04 22:27:06.750509-0800  localhost tccd[189]: [com.apple.TCC:access] AttributionChain: RESP:{ID: com.apple.Terminal, PID[934],
auid: 501, euid: 501, responsible path: '/Applications/Utilities/Terminal.app/Contents/MacOS/Terminal', binary path: '/Applications/
Utilities/Terminal.app/Contents/MacOS/Terminal'}, ACC:{ID: ??, PID[1232], auid: 501, euid: 501, binary path: '/private/tmp/espl'}, REQ:
{ID: com.apple.WindowServer, PID[169], auid: 88, euid: 88, binary path: '/System/Library/PrivateFrameworks/SkyLight.framework/Versions/A/
Resources/WindowServer'}
...
2019-11-04 22:27:06.773781-0800  localhost espl[1232]: (LaunchServices) [com.apple.launchservices:cas]
{ "ApplicationType"="BackgroundOnly", "CFBundleExecutablePath"="/private/tmp/espl", "CFBundlePackageType"="????",
"CFBundleSignature"="????", "Flavor"=2, "LSArchitecture"="x86_64", "LSCheckInTime*"=now-ish 2019/11/04 22:27:06, "LSDisplayName"="espl",
"LSExecutableFileName"="espl" }
...
2019-11-04 22:30:05.335596-0800  localhost espl[1275]: NSHomeDirectory() /var/root
2019-11-04 22:30:25.958905-0800  localhost espl[1275]: result = sudo systemsetup -getremotelogin
2019-11-04 22:30:25.959275-0800  localhost espl[1275]: running task sudo systemsetup -getremotelogin
2019-11-04 22:31:13.902658-0800  localhost espl[1275]: result = sudo systemsetup -setremotelogin on
2019-11-04 22:31:13.902753-0800  localhost espl[1275]: running task sudo systemsetup -setremotelogin on
2019-11-04 22:36:25.272792-0800  localhost espl[1275]: result = ifconfig
2019-11-04 22:36:25.272954-0800  localhost espl[1275]: running task ifconfig
2019-11-04 22:39:58.002521-0800  localhost espl[1314]: (libsystem_info.dylib) Created Activity ID: 0x7570, Description: Retrieve User by
ID
2019-11-04 22:39:58.015699-0800  localhost espl[1314]: NSHomeDirectory() /Users/john
2019-11-04 22:40:21.886909-0800  localhost opendirectoryd[71]: [com.apple.opendirectoryd:session] PID: 1314, Client: 'espl', exited with
0 session(s), 0 node(s) and 0 active request(s)
2019-11-04 22:40:22.049657-0800  localhost espl[1328]: NSHomeDirectory() /var/root
2019-11-04 22:40:25.539763-0800  localhost espl[1328]: result = ifconfig
2019-11-04 22:40:25.540021-0800  localhost espl[1328]: running task ifconfig
```

CONCLUSION

WHAT YOU HAVE LEARNED

- What the Unified Log is
- What its functions are
- How to acquire it
- Various tools, both native and third party
- How to narrow down your scope and find evil faster



AUTOMACTC MODULE RELEASE

- Upcoming AutoMacTC module release will include unified log parsing features
  - We are working on characterizing the log show bug before we release the module

https://github.com/CrowdStrike/automactc

CROWDSTRIKE

THANK YOU

ANY QUESTIONS?