

Unified Logging



Richard Mallion

dataJAR Ltd

Email: richard@datajar.co.uk

Twitter: [@richardmallion](https://twitter.com/richardmallion)

MacAdmins Slack: [@red5](https://macadmins.slack.com)



Unified Logging



Overview

Log Levels

Subsystems and Categories

Log Command

Predicates



Traditional Logs



Introduced at WWDC 2016

Centralised logging

Standardised logging format

Maximise information collected

Small footprint

Compression of log messages

Defer lot of the work until observation

Manage log message life cycle

Privacy by design



Unified Logging



Supported Platforms

macOS 10.12+

iOS 10+

tvOS 10+

watchOS 3+





New File Format

.tracev3 files

Compressed binary format

/var/db/diagnostics

/var/db/diagnostics/persist

Approx 52 files are maintained

Each approx 10.5MB

/var/db/uuidtext



Must Use Apple Tools

Console.app

log command line tool

Xcode

OSLog framework



How Does it Work?





Log Levels for
Each Message

Each message has a log level





Log Levels for
Each Message

Three basic levels

- Default (Notice) - Essential for troubleshooting
- Info - Helpful but not essential for troubleshooting
- Debug - Useful during debugging





Log Levels for
Each Message

Two special levels

- **Error** - Errors seen during the execution of your code
- **Fault** - Faults and bugs in your code





Subsystems and Categories

Each log message is associated with a

- subsystem
- category





Subsystems

Subsystems identify an app or part of an app

Uses reverse-DNS Notation

com.example.myapp





Categories

Categories identify a particular component or module in a given subsystem

Which can be used to filter logs

Network, start-up, authentication....





Subsystems and Categories

<i>Subsystem</i>	<i>Category</i>
<i>uk.co.datajar.projectx</i>	<i>Network</i>
<i>uk.co.datajar.projecty</i>	<i>login, password</i>



Basic Log Level characteristics

- Each log level can have two characteristics
- Can be set for system, subsystem or category





Basic Log Level
characteristics

Is it Enabled?
Stored in disk or memory





Default behaviour -
characteristics

<i>Message Level</i>	<i>Enabled</i>	<i>Destination</i>
<i>Default (Notice)</i>	<i>Always</i>	<i>Disk</i>
<i>Info</i>	<i>Yes</i>	<i>Memory</i>
<i>Debug</i>	<i>No</i>	<i>N/A</i>
<i>Error</i>	<i>Always</i>	<i>Disk</i>
<i>Fault</i>	<i>Always</i>	<i>Disk</i>



Default behaviour - characteristics

You can check the current log level of a subsystem
`sudo log config --status --subsystem uk.co.datajar.myapp`

Mode for 'uk.co.datajar.myapp' INFO PERSIST_DEFAULT

You can enable debug logging for a subsystem

```
sudo log config --mode "level:debug" --subsystem  
uk.co.datajar.myapp
```

Creates a plist in /Library/Preferences/Logging/Subsystems



Default behaviour -
characteristics

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">  
<plist version="1.0">  
<dict>  
    <key>DEFAULT-OPTIONS</key>  
    <dict>  
        <key>Level</key>  
        <dict>  
            <key>Enable</key>  
            <string>debug</string>  
        </dict>  
    </dict>  
</dict>  
</plist>
```



You can enable persistence debug logging for a subsystem

```
sudo log config --mode "level:debug, persist:debug"  
--subsystem uk.co.datajar.myapp
```

Creates a plist in /Library/Preferences/Logging/Subsystems



Default behaviour -
characteristics

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">  
<plist version="1.0">  
<dict>  
    <key>DEFAULT-OPTIONS</key>  
    <dict>  
        <key>Level</key>  
        <dict>  
            <key>Enable</key>  
            <string>debug</string>  
            <key>Persist</key>  
            <string>debug</string>  
        </dict>  
    </dict>  
</dict>  
</plist>
```





Default behaviour -
characteristics

Apple have a set of overrides for their various
subsystems.

[`/System/Library/Preferences/Logging/Subsystems`](#)



com.apple.su.plist

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
  <dict>
    <key>SU</key>
    <key>DEFAULT-OPTIONS</key>
    <dict>
      <key>Level</key>
      <dict>
        <key>Enable</key>
        <string>Info</string>
        <key>Persist</key>
        <string>Info</string>
      </dict>
      <key>TTL</key>
      <dict>
        <key>Default</key>
        <integer>30</integer>
        <key>Notice</key>
        <integer>30</integer>
        <key>Info</key>
        <integer>30</integer>
        <key>Debug</key>
        <integer>30</integer>
      </dict>
    </dict>
  </dict>
</plist>
```

Default behaviour -
characteristics



You can also override the logging level with a plist / configuration profile.

`/Library/Preferences/Logging/Subsystems`



Default behaviour - characteristics

https://developer.apple.com/documentation/os/logging/customizing_logging_behavior_while_debugging



log



log

A terminal window titled 'richard — -bash — 107x28'. The window contains the following text:

```
[macbook:~ richard$ log
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
macbook:~ richard$ ]
```

log stream – Live logs



log

```
[macbook:~ richard$ log stream
Timestamp           Thread   Type      Activity          PID    TTL
2021-06-11 16:00:55.320874+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:png typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.461385+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:jpg typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.511616+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:jpg typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.561030+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:jpg typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.611822+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:jpg typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.660972+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:jpg typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.711295+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:jpg typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.761926+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:png typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.814388+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:png typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.862062+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:png typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.911922+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:jpg typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:55.962468+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:jpg typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:56.012184+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:png typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:56.062063+0100 0xd3ae7c  Default   0x0           148    0  revisiond: (ChunkingLibrary) [com.apple.chunkinglibrary=default] ty
peHint:jpeg typeRequested:kCKProfileTypeFixed typeUsed:kCKProfileTypeFixed sectionCount:1 default:F resolvedType:(null) serverConfig:(null)
2021-06-11 16:00:56.122339+0100 0xd3b2ee  Activity  0x23d7a3d       68660  0  GoogleSoftwareUpdateDaemon: (Security) SecTrustEvaluateIfNecessary
2021-06-11 16:00:56.122359+0100 0xd3b134  Activity  0x23d7a3f       68660  0  GoogleSoftwareUpdateDaemon: (Security) SecTrustEvaluateIfNecessary
2021-06-11 16:00:56.122382+0100 0xd3b2f0  Activity  0x23d7a50       68660  0  GoogleSoftwareUpdateDaemon: (Security) SecTrustEvaluateIfNecessary
2021-06-11 16:00:56.122388+0100 0xd3b2f1  Activity  0x23d7a3e       68660  0  GoogleSoftwareUpdateDaemon: (Security) SecTrustEvaluateIfNecessary
2021-06-11 16:00:56.122508+0100 0xd3b2ef  Activity  0x23d7a51       68660  0  GoogleSoftwareUpdateDaemon: (Security) SecTrustEvaluateIfNecessary
2021-06-11 16:00:56.122725+0100 0xd3b2ec  Activity  0x23d7a52       68660  0  GoogleSoftwareUpdateDaemon: (Security) SecTrustEvaluateIfNecessary
```

log show – Historical logs



log

```
[macbook:~ richard$ log show
Skipping info and debug messages, pass --info and/or --debug to include.
Timestamp           Thread   Type     Activity      PID    TTL
2021-05-24 11:21:02.856136+0100 0x0    Timesync  0x0          0      0    === log class: TTL more than 14 days begins
2021-05-24 21:37:55.092132+0000 0x0    Timesync  0x0          0      0    === system boot: DABA53C8-718B-4452-BBB4-64430EF67C18
2021-05-24 21:38:11.452295+0000 0x334  Default   0x0          76     3    bluetoothd: [com.apple.bluetooth:bluetoothd] DaemonSetupPCIeTrans
sportHostController BTI showed up
2021-05-24 21:38:11.473626+0000 0x333  Fault     0xa0         75     14    recoveryosd: (RecoveryOS) <private>
2021-05-24 21:38:11.487138+0000 0x334  Error     0x0          76     3    bluetoothd: [com.apple.bluetooth:bluetoothd] DaemonSetupPCIeTrans
sportHostController unable to find io service
2021-05-24 21:38:11.491617+0000 0x333  Fault     0xa1         75     14    recoveryosd: (RecoveryOS) <private>
2021-05-24 21:38:11.529955+0000 0x334  Error     0x0          76     3    bluetoothd: (IOBluetooth) [com.apple.bluetooth:bluetoothd] No Bl
uetooth preference file
2021-05-24 21:38:11.546368+0000 0x334  Fault     0xb1         76     14    bluetoothd: (CoreFoundation) [com.apple.defaults:User Defaults]
Couldn't write values for keys (
    DeviceCache
) in CFPrefsPlistSource<0x7ffb85c05960> (Domain: com.apple.Bluetooth, User: kCFPreferencesAnyUser, ByHost: Yes, Container: (null), Contents Need Refresh
: No): setting preferences outside an application's container requires user-preference-write or file-write-data sandbox access
2021-05-24 21:38:11.547831+0000 0x334  Fault     0xb2         76     14    bluetoothd: (CoreFoundation) [com.apple.defaults:User Defaults]
Couldn't write values for keys (
    PersistentPortsServices
) in CFPrefsPlistSource<0x7ffb85c05960> (Domain: com.apple.Bluetooth, User: kCFPreferencesAnyUser, ByHost: Yes, Container: (null), Contents Need Refresh
: No): setting preferences outside an application's container requires user-preference-write or file-write-data sandbox access
2021-05-24 21:38:11.548651+0000 0x334  Fault     0xb3         76     14    bluetoothd: (CoreFoundation) [com.apple.defaults:User Defaults]
Couldn't write values for keys (
    PersistentPorts
) in CFPrefsPlistSource<0x7ffb85c05960> (Domain: com.apple.Bluetooth, User: kCFPreferencesAnyUser, ByHost: Yes, Container: (null), Contents Need Refresh
: No): setting preferences outside an application's container requires user-preference-write or file-write-data sandbox access
2021-05-24 21:38:11.567106+0000 0x334  Fault     0xb4         76     14    bluetoothd: (CoreFoundation) [com.apple.defaults:User Defaults]
Couldn't write values for keys (
    BluetoothVersionNumber
) in CFPrefsPlistSource<0x7ffb85c05960> (Domain: com.apple.Bluetooth, User: kCFPreferencesAnyUser, ByHost: Yes, Container: (null), Contents Need Refresh
: No): setting preferences outside an application's container requires user-preference-write or file-write-data sandbox access
2021-05-24 21:38:11.762006+0000 0x334  Fault     0xb5         76     14    bluetoothd: (CoreFoundation) [com.apple.defaults:User Defaults]
```



log

By default info and debug messages are skipped

`log show --info`

`log show --debug`





Filtering

Log





Date and Time constraints

log





log

Show events relative to the end of the logs

log show --last 1

log show --last 1m

log show --last 1h

log show --last 1d

log show --last boot





Show events from a start date

Format YYYY-MM-DD HH:MM:SS

log show --start '2021-06-23 11:30:00'

log





Show events to an end date

Format YYYY-MM-DD HH:MM:SS

log show --end '2021-06-23 11:30:00'

log





log

Combine start and end dates

```
log show --start '2021-06-23 11:30:00' --end  
'2021-06-23 11:31:00'
```





Predicate based filtering

We can use predicates to filter on subsystems, categories and other data

log



Predicate - subsystem

log show



Predicate - subsystem

log show --predicate

Predicate - subsystem

```
log show --predicate '
```



Predicate - subsystem

```
log show --predicate '
```

Predicate - subsystem

```
log show --predicate 'subsystem == "'
```



Predicate - subsystem

```
log show --predicate 'subsystem == "uk.co.datajar.psumaclogger"'
```



Predicate - category

```
log show --predicate '
```



Predicate - category

```
log show --predicate 'category == " "'
```



Predicate - category

```
log show --predicate 'category == "network"'
```



Predicate - subsystem and category

```
log show --predicate '
```



Predicate - subsystem and category

```
log show --predicate 'subsystem == "uk.co.datajar.psumaclogger"  
'
```



Predicate - subsystem and category

```
log show --predicate 'subsystem == "uk.co.datajar.psumaclogger"  
and  
'
```



Predicate - subsystem and category

```
log show --predicate 'subsystem == "uk.co.datajar.psumaclogger"  
and category == "network"'
```



Predicate - subsystem and category with time constraint

```
log show --predicate 'subsystem == "uk.co.datajar.psumaclogger"  
and category == "network"' --last 5m
```



Predicate - subsystem and category with time constraint

```
log show --predicate 'subsystem == "uk.co.datajar.psumaclogger"  
and category == "network"' --start '2021-06-23 11:30:00' --end  
'2021-06-23 11:31:00'
```



Predicate - eventMessage

```
log show --predicate '
```



Predicate - eventMessage

```
log show --predicate 'eventMessage contains "foobar"'
```



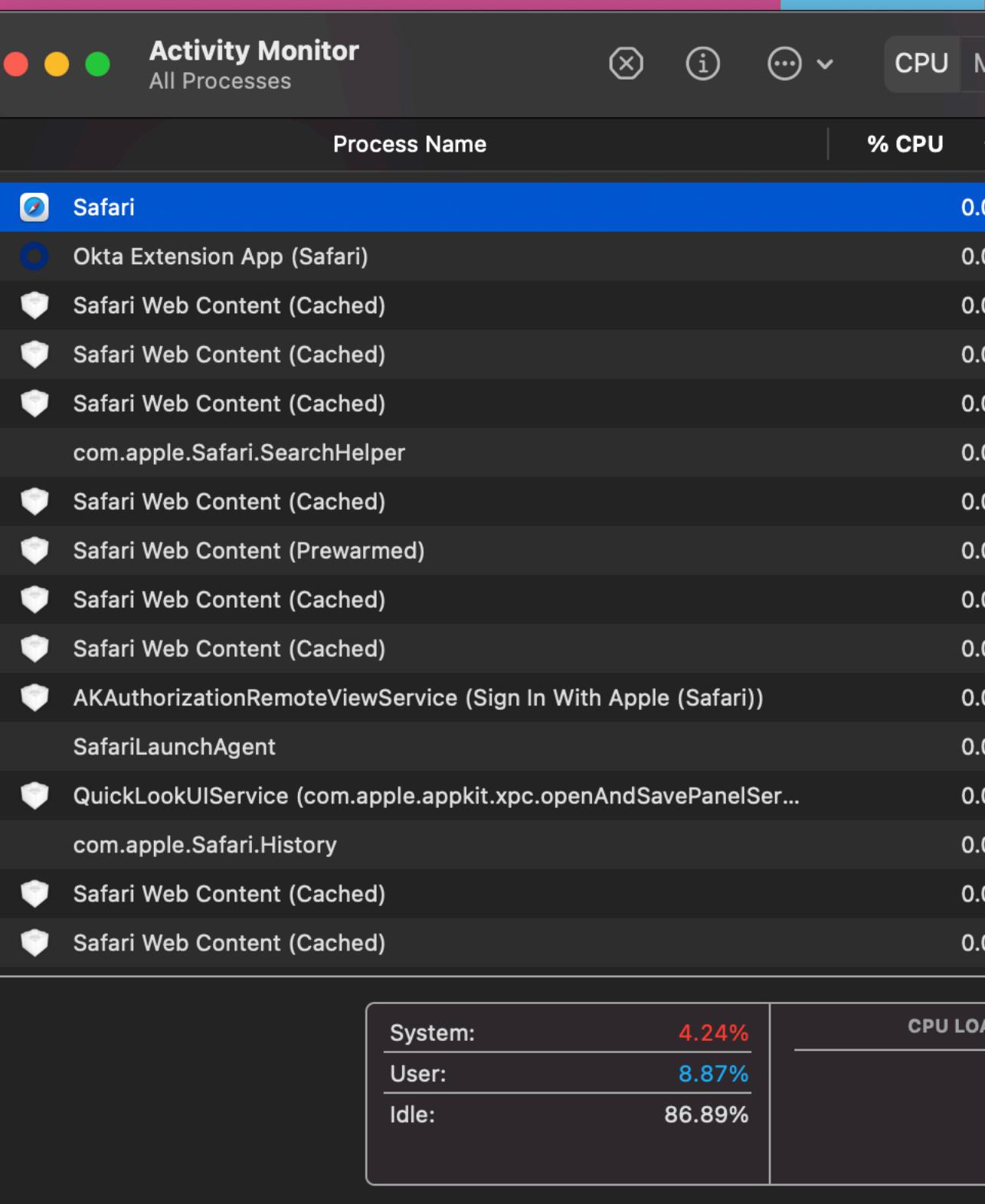
Predicate - subsystem, category and message

```
log show --predicate 'subsystem == "uk.co.datajar.psumaclogger"  
and category == "network" and eventMessage contains "foobar"'
```



Predicate - process (the process that originated the event)

```
log show --predicate 'process == "Safari"'
```



Predicate - processImagePath (The full path of the process that originated the event)

```
log show --predicate 'process == "sudo"'  
//Results
```

```
log show --predicate 'processImagePath == "sudo"'  
//No Results
```

```
log show --predicate 'processImagePath contains "sudo"'  
//Results
```

```
log show --predicate 'processImagePath == "/usr/bin/sudo"'  
//Results
```



Predicate - sender (The name of the library, framework, kernel extension that originated the event)

Predicate - senderImagePath (The full path of the library, framework, kernel extension that originated the event)

Predicate - or

```
log show --predicate 'process == "Safari" or process == "Mail"'
```



Predicate - Grouping your logic

```
log show --predicate 'process == "su" or process == "sudo" and  
eventMessage contains "tty"'
```



Predicate - Grouping your logic

```
log show --predicate 'process == "su" or process == "sudo" and  
eventMessage contains "tty"'
```



Predicate - Grouping your logic

```
log show --predicate 'process == "su" or process == "sudo" and  
eventMessage_contains "tty"'
```



Predicate - Grouping your logic

```
log show --predicate '(process == "su" or process == "sudo") and  
eventMessage contains "tty"'
```



Predicate - case issues

```
log show --predicate 'process == "SaFaRi"'
```



Predicate - case insensitive

```
log show --predicate 'process ==[ c ] "SaFaRi" '
```



Predicate - case insensitive

```
log show --predicate 'process contains[c] "SaFaRi"'
```



Predicate - beginswith

```
log show --predicate 'process beginswith[ c ] "SaF"'
```



Predicate - endswith

```
log show --predicate 'process endswith[c] "Ari"'
```





Other Comparisons

<i>Operator</i>	<i>Description</i>
= ==	<i>Equal</i>
!= <>	<i>Not equal</i>
>= =>	<i>Greater than or equal</i>
<= =<	<i>Less than or equal</i>
>	<i>Greater than</i>
<	<i>Less than</i>
AND &&	<i>Logical AND</i>
OR	<i>Logical OR</i>
NOT !	<i>Logic NOT</i>



Other Comparisons

<i>Operator</i>	<i>Description</i>
BEGINSWITH	<i>The left-hand expression begins with the right-hand expression</i>
CONTAINS	<i>The left-hand expression contains the right-hand expression</i>
ENDSWITH	<i>The left-hand expression ends with the right-hand expression</i>
LIKE	<i>The left hand expression equals the right-hand expression: .? and .* are allowed as wildcard characters</i>
MATCHES	<i>The left hand expression equals the right hand expression using a regex-style comparison</i>





Predicate - Documentation

<https://developer.apple.com/library/mac/documentation/Cocoa/Conceptual/Predicates/Articles/pSyntax.html>

log



Predicate - messageType

default (0) , info (1) , debug (2) , error (16) , fault (17)

log show --predicate 'messageType == "info"'



Log Archives



Can export yours logs to a `.logarchive`

Can specify last or start predicates

```
sudo log collect --last 5m --output ~/Desktop  
log
```



Log Archives



log

Can open in Console App

Can also use

`log show ~/Desktop/mylogs.logarchive`



Collecting from other paired devices

--device-name

--device-udid

```
sudo log collect --device-name iPad --last 5m --output ~/Desktop
```



Predicate - Log Styles

--style default, compact, json, ndjson, syslog



Privacy



We don't want personal data leaking

Data that is dynamic is considered private

Collections of data (Arrays) are considered private

Privacy

PSUMAC Logger: [uk.co.datajar.psumaclogger:password] Password is <private>



Privacy

For macOS 10.12 - 10.14, redaction can be disabled

`sudo log config --mode 'private_data:on'`

`sudo log config --mode 'private_data:off'`

log





Privacy

For macOS 10.15.3+ a configuration profile can be used

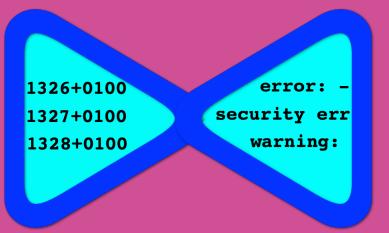
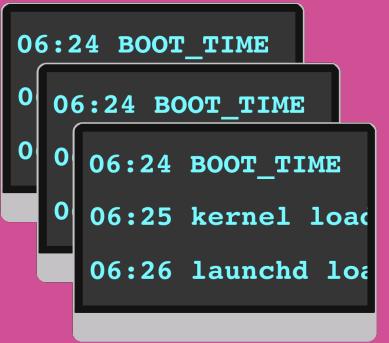
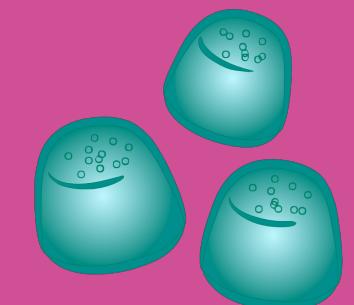
<https://georgegarside.com/blog/macos/sierra-console-private/>

log





The Eclectic Light
Company



Ulbow: Easy to use Log Browser

Consolation: Log Browser

Mints: Log toolbox

Blowhole : Log from command line

<https://eclecticlight.co>



<https://github.com/dataJAR/JNUC2021-UnifiedLogging>