< Back to Mann Documentation Index

Mann Jamf Workflows
Jamf Client Communications Doctor

Summary

Jamf Pro doesn't have guard rails in place for when it runs policies or inventory updates to ensure communications remain stable. Of primary concern is the `jamf` process doesn't place time limits on how long a policy or extension attribute is allowed to run. In addition to this all tasks are run sequentially. If a single process hangs this will cause all other pending tasks to wait indefinitely for it to complete, which may never happen. In the Jamf Pro Server console this may reflect computers not properly having inventory updated or will only run some policies on an infrequent basis. Additionally, Jamf may still list the computer as "checking in," however there may usually be little to no policy logs.

Some common examples of objects that will cause a hang:

- A script that waits for user input.

- A script that implements Jamf Helper without the -timeout flag.
- A script that runs a child process which subsequently hangs.
- An installer package waits for input or external data.
- The computer goes to sleep during the check-in process.

For Mann's Jamf MSP licensed Customers Mann will remotely be alerted via Datadog of the action taken as well as provided with the computer ID in Jamf and the child process that was killed. This allows for quick identification and remediation of bad workflows in Jamf.

Employee experience

There is no employee experience, all action is transparent and occurs in the background.

Features and Settings

All settings are enforced using a configuration profile with the custom domain of `com.mann.JamfClientCommunicationsRepair` - a sample configuration profile has been uploaded to your instance in a disabled state to edit later.

Policy Kill Threshold

Value: POLICY_KILL_THRESHOLD
Type: Integer
Default: 850
The default threshold for killing a policy is 850 minutes, this is intentionally slightly more than the TTL for Mann's longest running workflow, our macOS Upgrade workflow. Mann does not recommend using a value lower 850 this to avoid accidentally killing a critical macOS upgrade.

Configuration: *Edit the configuration profile "Jamf Client Communications Repair Config" under the payload Application & Custom Settings. Change the value for POLICY_KILL_THRESHOLD, setting the number of minutes you wish to allow policy processes to run.*

Extension Attribute Threshold

Value: EA_KILL_THRESHOLD

Type: Integer
Default: 60
The default threshold for killing an extension attribute is 60 minutes, this should be more than enough time for any data processing or collection to occur.

Configuration: *Edit the configuration profile "Jamf Client Communications Repair Config" under the payload Application & Custom Settings. Change the value for EA_KILL_THRESHOLD, setting the number of minutes you wish to allow extension attributes to run.*

Caffeinate On AC Power

Value: caffeinateJamf
Type: Boolean
Default: true
This will prevent the system from sleeping (caffeinated) when on AC power while the `jamf` Process is alive. This will not prevent normal display sleep or prevent sleep while on battery.

Configuration: *Edit the configuration profile "Jamf Client Communications Repair Config" under the payload Application & Custom Settings and set caffeinateJamf to True.*

Caffeinate Timer

Value: caffeinateTimer
Type: Integer
Default: 54834
This will define the maximum time the system will be prevented from sleeping (`caffeinate -s -t ${caffeinateTimer} -w ${jamfPid} &`) when on AC power while the `jamf` process is alive. This will not prevent normal display sleep or prevent sleep while on battery. This caffeinates the `jamf` process, the system will remain caffeinated until either the `jamf` process exits or the timer runs out, whichever comes first.

Configuration: *Edit the configuration profile "Jamf Client Communications Repair Config" under the payload Application & Custom Settings and set caffeinateTimer as an integer to the max number of seconds caffeinate is allowed.*

Datadog Logging

For Mann's Jamf MSP Licensed Customers this workflow utilizes Mann's centralized Datadog logging to provide alerting on errors and diagnostics data. More information can be found in our Mann Common Functions - Datadog Logging documentation.

Installation

Mann Jamf MSP Licensed Customers

For Jamf MSP Customers this workflow will be uploaded automatically along with the Jamf Client Communications suite.

Non Mann Jamf MSP Customers

Policy

1. Download a copy of Jamf Client Communications Doctor.sh
2. Navigate to Jamf Pro > Settings > Scripts and create a new Script
    a. Display Name: Jamf Client Communications Doctor
    b. Script: Paste in the contents of Jamf Client Communications Doctor.sh
3. Save your Script
4. Create a new policy
    a. **Trigger:** Check-in and Network State Change
    b. **Frequency:** Weekly
    c. Add Jamf Client Communications Doctor in the Script payload
    d. Scope to the computers you want installed

Create a new Extension Attribute

1. Navigate to **Settings** > **Computer management** > **Extension Attributes**
2. Click **+ New**
    a. **Display Name**: Jamf Client Communications Doctor Last Error
    b. **Enabled**: Yes
    c. **Data Type**: String
    d. **Inventory Display**: Extension Attributes
    e. **Input Type**: Script
        i. Paste contents of EA from Github
3. Click Save

Testing

The following methods will allow you to test the various functions that will cause the Jamf Client Communications Doctor to take action to protect Jamf communications for your computers.

Test Blocking Installer

A blocking installer should be executed locally for proper testing. Steps to execute a blocking installer:

1. Download Blocking Installer Package.pkg to your computer
2. Open Terminal and run: `sudo installer -pkg ./"Blocking Installer Package.pkg" -target /`
3. Leave the terminal window open and running for over 2 days.
4. Watch `/var/log/JamfClientCommunicationsDoctor.log`, you'll see a line similar to:

```
None
```

Test Blocking Policy

To create a blocking policy to test take the following steps:

1. Navigate to Jamf Pro > Settings > Scripts and create a new Script
    a. Display Name: Jamf Client Communications Doctor Test Script
    b. Enabled: Yes
    c. Script:

```
None
#!/bin/zsh
# Blocking Script will live for more than Mann's Jamf Client Communications Doctor's default 850 minutes TTL
# More information at https://mann.com/jamf/doctor
for i in {1..16}; do sleep 3600; done
```

2. Scope a computer to a policy and wait, the default timer is 14.1 hours so
    a. Do not use `sudo jamf recon` for testing, the execution should happen naturally.
3. Watch `/var/log/JamfClientCommunicationsDoctor.log`, you'll see a line similar to:

```
None
2023-08-18 10:32:27 ERROR : https://company.jamfcloud.com/ : JamfClientCommunicationsDoctor : 20230810 : 0-20591 : Child policy at 81065 has been running for longer than 850 minutes. Killing the policy. First 10 lines of /Library/Application Support/JAMF/tmp/Jamf Client Communications Doctor Test Script for debugging:
```

```
#!/bin/zsh
# Blocking Script will live for more than Mann's Jamf Client Communications Doctor's
default 850 minutes TTL
# More information at https://mann.com/jamf/doctor
for i in {1..16}; do sleep 3600; done

Child Processes:
PID TTY      TIME CMD
81066 ??     0:00.01 /bin/zsh /Library/Application Support/JAMF/tmp/Jamf Client
Communications Doctor Test Script / MacBook Pro administrator
81068 ??     0:00.00 sleep 3600
```

Test Blocking Extension Attribute

A blocking Extension Attribute will run on all computers at recon, any computers that don't have the Jamf Client Communications Doctor running may no longer check-in.

To create a blocking extension attribute to test take the following steps:

4. Navigate to Jamf Pro > Settings > Computer Extension Attributes and create a new Extension Attribute
    a. Display Name: Jamf Client Communications Doctor Test
    b. Enabled: Yes
    c. Data Type: String
    d. Input: Script

```
None
#!/bin/zsh
# Blocking Extension Attribute that will live for more than Mann's Jamf Client
Communications Doctor's 60 minute TTL
# More information at https://mann.com/jamf/doctor
for i in {1..80}; do sleep 60; done
```

5. Scope the computer to a policy that has a Inventory Update OR wait for the next daily inventory update
    a. Do not use `sudo jamf recon` for testing, the execution should happen naturally.
6. Watch `/var/log/JamfClientCommunicationsDoctor.log`, you'll see a line similar to:

None

```
2023-08-17 15:14:29 ERROR : https://company.jamfcloud.com/ :
JamfClientCommunicationsDoctor : 20230810 : 0-10002 : Child extension-attribute
at 26883 has been running for longer than 60 minutes. Killing the
extension-attribute. First 10 lines of /Library/Application
Support/JAMF/tmp/21C9A9A9-A1E5-4C92-BFFF-138C8A9A79DE for debugging:

#!/bin/zsh
# Blocking Extension Attribute that will live for more than Mann's Jamf Client
Communications Doctor's default 60 minute TTL
# More information at https://mann.com/jamf/doctor
for i in {1..80}; do sleep 60; done

Child Processes:
PID TTY           TIME CMD
28015 ??         0:00.00 sleep 60
```