

A Deep Dive Into Apple OS Updates

(now with more DDM!)

Hello!



Bryce Carlson
Product Manager



Manny Cabrera
Solutions Architect

The background is dark gray with horizontal dashed lines. Scattered throughout are small, short dashes in orange, teal, and white, some of which are slightly tilted.

99.9%

Of your devices are in end user hands

The background is dark gray with horizontal dashed lines. Scattered throughout are small, short dashes in orange, cyan, and white, some of which are slightly tilted.

100%

Of your devices are a threat vector



System Updates via MDM

System Updates via MDM



Agenda

- Updates via MDM (what, how, why)
- MDM Protocol for Updates
- MDM OS Updates user experience
- Updates via DDM improvements
 - Why are they needed
- MDM Watchdog
- macOS Updates via Script/Public Software
- Q/A

System Updates via MDM

What

System Updates installed on devices via MDM

How

Device Channel MDM commands sent to the device

Why?

macOS 11+ and more in macOS 12+



Requirements

Device is Supervised

macOS 12 and newer

ADE or Reduced Security Mode

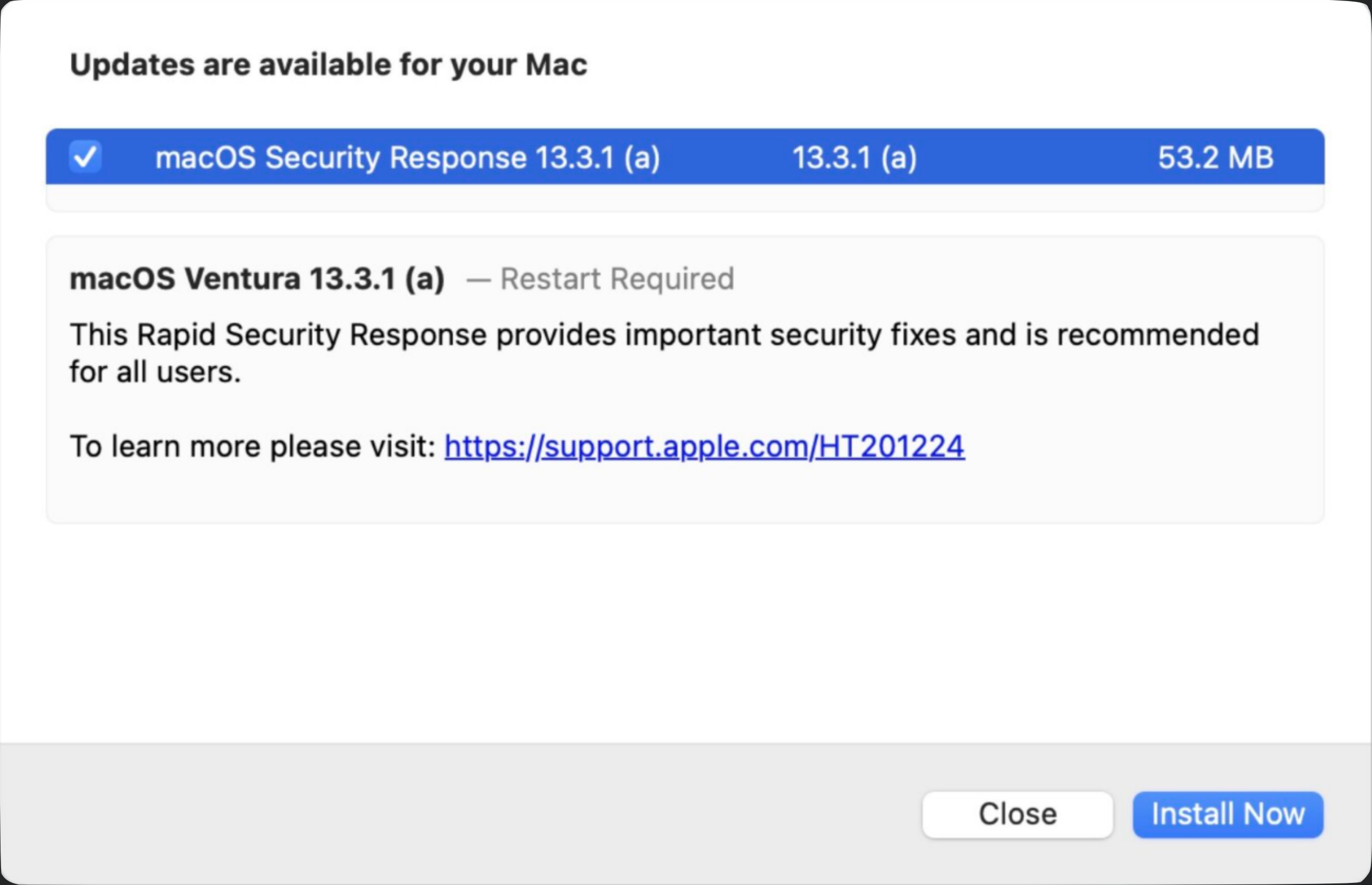
iOS 9 and newer

iPadOS 13 and newer

tvOS 12 and newer



System Updates via MDM



System Updates via MDM

System Updates Status

Some devices may need to be restarted to clear the current Update Status.

Update StatusAvailable UpdatesHistory

Update available: XProtectPlistConfigDataVersion: 2167Last update: May 2, 2023 12:30:23PM EDT

✓ Update command sent & received

Installation started

Downloaded

Installing

Update available: XProtectPayloadsVersion: 95Last update: May 2, 2023 12:30:23PM EDT

✓ Update command sent & received

Installation started

Downloaded

Installing

Update available: macOS Security Response 13.3.1 (a)Version: 13.3.1Last update: May 2, 2023 12:38:33PM EDT

✓ Update command sent & received

Installation started

Downloaded

Installing

System Updates Status

Device: iPhone

Current OS: 16.4.1

Supervised

 Refresh

Some devices may need to be restarted to clear the current Update Status.

Update Status


Available Updates

History

Update available: iOS Security Response 16.4.1 (a)

Version: 16.4.1

Last update: May 2, 2023 11:34:56AM CDT

 Update command sent & received

Installation started

Downloaded

Installing

System Updates via MDM

- Keep updated or set max version
- Re-send command
- Enforce when available, or allow user to defer
- Schedule updates to avoid user interruptions

☒ **Enable macOS Updates**

☒ Set maximum version13.99.99

☐ Keep devices updated to the latest OS (including major versions)

☒ Re-send update command if last status is older than: 8

The default time for this property is set to 24 hours.

OptionsDefault

Download and install depending on current device state.

☒ **Enable iOS Updates**

☐ Set maximum version15.6

☒ Keep devices updated to the latest OS (including major versions)

☐ Re-send update command if last status is older than: 24

The default time for this property is set to 24 hours.

☒ **Enable iPadOS Updates**

☐ Set maximum version15.6

☒ Keep devices updated to the latest OS (including major versions)

☐ Re-send update command if last status is older than: 24

The default time for this property is set to 24 hours.

☐ **Enable tvOS Updates**

☒ Set maximum version15.6

☐ Keep devices updated to the latest OS (including major versions)

☐ Re-send update command if last status is older than: 24

The default time for this property is set to 24 hours.

☒ **Schedule Updates**

Set the days and times when update commands are sent. Start time is based on each device's current time. The default schedule is nightly at 2am UTC.

Days Allowed

☒ Sun☐ Mon☐ Tue☐ Wed☐ Thu☒ Fri☒ Sat

Start Time18:00Time Window4 hours

Stop Sending Commands30 min prior to the time window close.

InstallAction Options

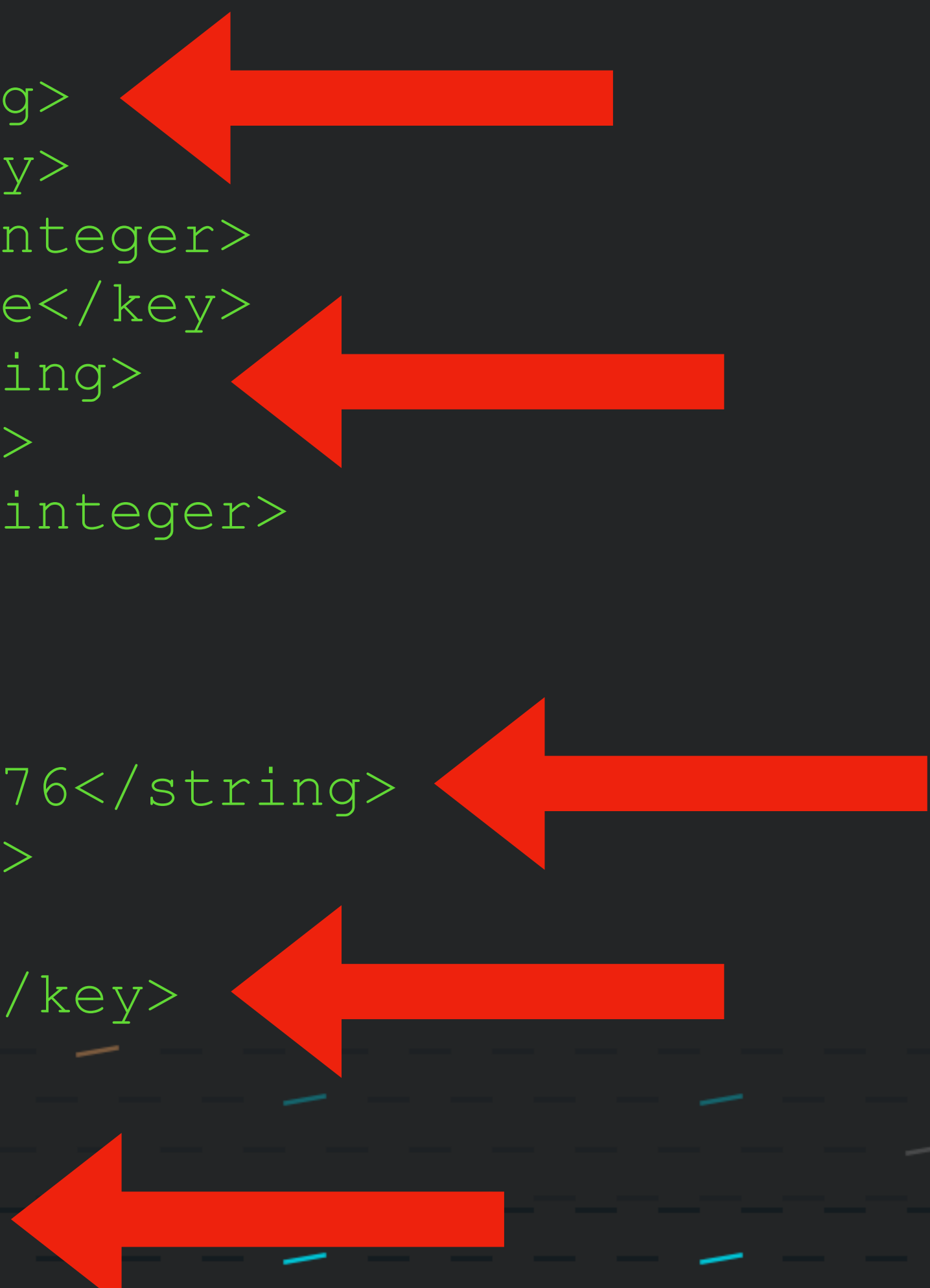
- *Default*
 - Download or install the update or upgrade, depending on the current state.
 - End user will get 60 second count down in Notification Center if a reboot is needed
- *InstallForceRestart*
 - Perform the default action, and then force a restart if the update requires it.
- *InstallLater* (this option supports end user deferrals)
 - Download the software update or upgrade and install it at a later time.
 - With **Deferrals allowed** set, the system will prompt the user once a day, up to the maximum amount of times, before showing the reboot pending (in Notification Center just like *Default* option) and having the device to continue with the minor update.
 - If "Allow user to defer minor updates" is not selected, the user will be able to infinitely defer updates.

AvailableOSUpdate (response)

```
<?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>AvailableOSUpdates</key>
  <array>
    <dict>
      <key>AllowsInstallLater</key>
      <false/>
      <key>Build</key>
      <string>17A576</string>
      <key>DownloadSize</key>
      <integer>251607570</integer>
      <key>HumanReadableName</key>
      <string>iOS 13.0</string>
      <key>InstallSize</key>
      <integer>1809842176</integer>
      <key>IsCritical</key>
      <false/>
      <key>ProductKey</key>
      <string>iOSUpdate17A576</string>
      <key>ProductName</key>
      <string>iOS</string>
      <key>RestartRequired</key>
      <true/>
      <key>Version</key>
      <string>13.0</string>
    </dict>
  </array>
  <key>CommandUUID</key>
  <string>0001_AvailableOSUpdates</string>
  <key>Status</key>
  <string>Acknowledged</string>
  <key>UDID</key>
  <string>00008020-000915083C80012E</string>
</dict>
</plist>
```


AvailableOSUpdate (response)

```
<key>AvailableOSUpdates</key>
  <array>
    <dict>
      <key>AllowsInstallLater</key>
      <false/>
      <key>Build</key>
      <string>17A576</string>
      <key>DownloadSize</key>
      <integer>251607570</integer>
      <key>HumanReadableName</key>
      <string>iOS 13.0</string>
      <key>InstallSize</key>
      <integer>1809842176</integer>
      <key>IsCritical</key>
      <false/>
      <key>ProductKey</key>
      <string>iOSUpdate17A576</string>
      <key>ProductName</key>
      <string>iOS</string>
      <key>RestartRequired</key>
      <true/>
      <key>Version</key>
      <string>13.0</string>
    </dict>
```



The diagram illustrates the structure of the AvailableOSUpdate response. Red arrows point to the following elements:

- `<string>17A576</string>` (Build)
- `<string>iOS 13.0</string>` (HumanReadableName)
- `<string>iOSUpdate17A576</string>` (ProductKey)
- `<string>iOS</string>` (ProductName)
- `<string>13.0</string>` (Version)

ScheduleOSUpdate (send)

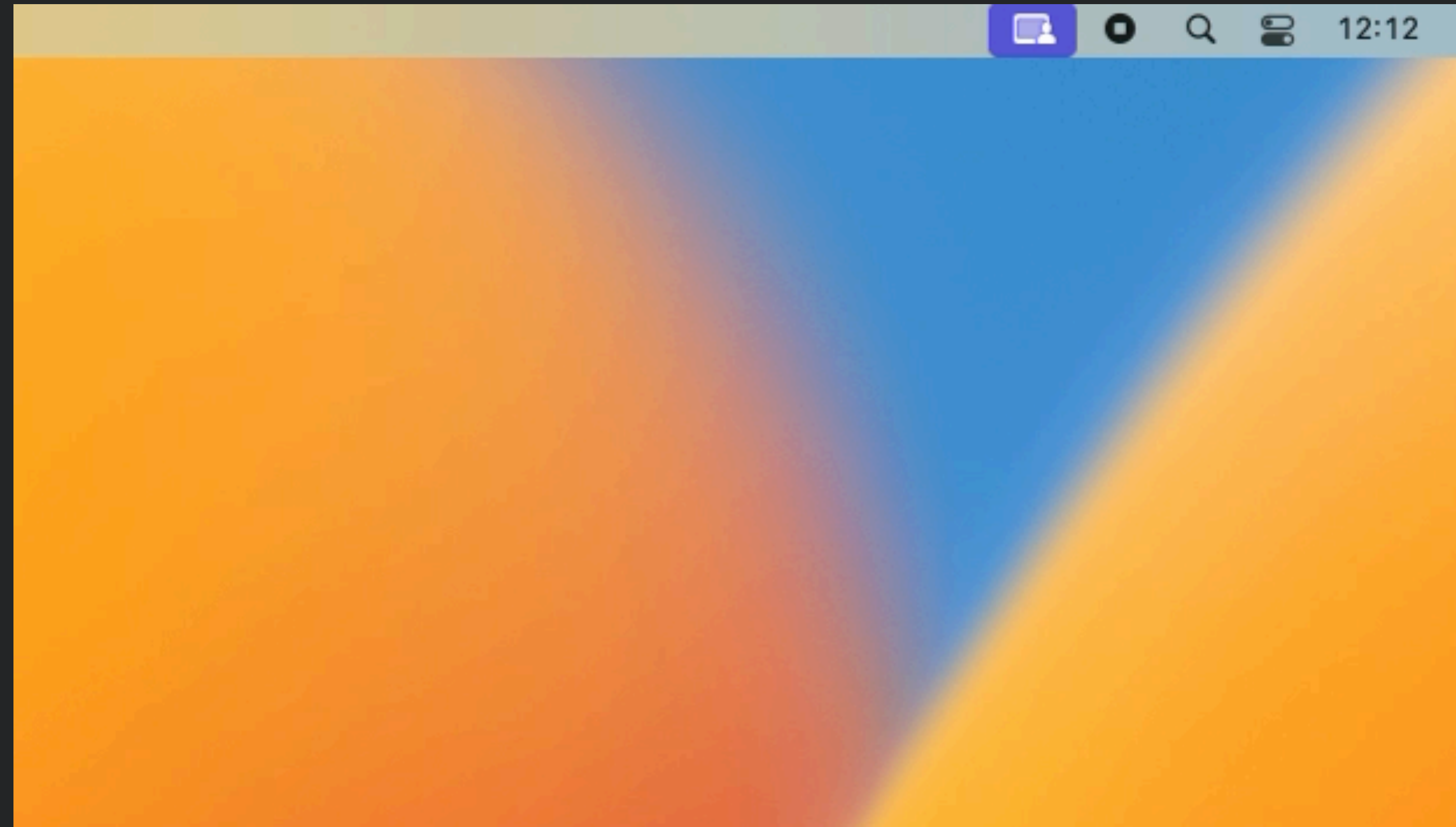
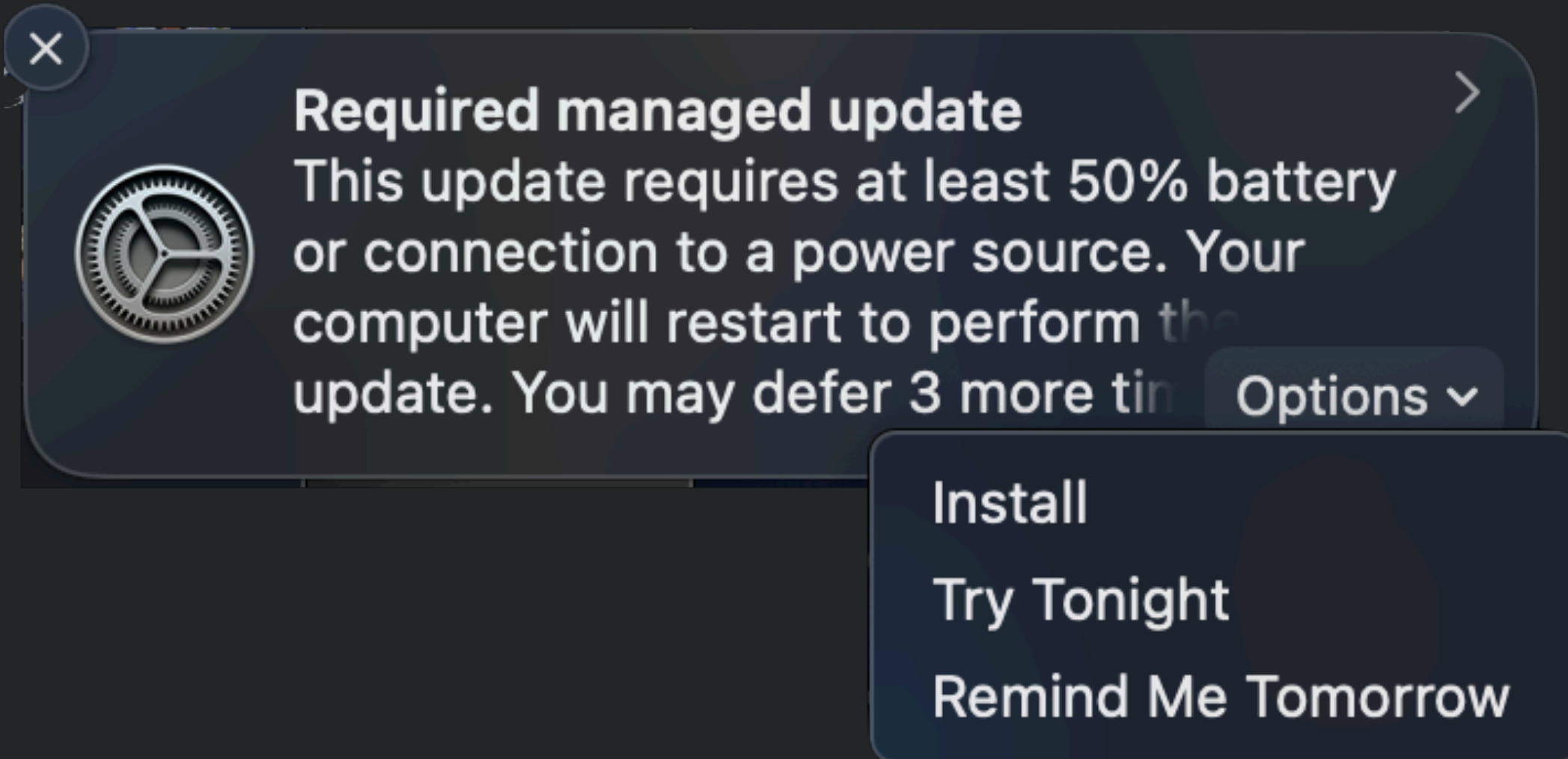
```
<?xml version="1.0" encoding="UTF-8"?>
  <!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
  <plist version="1.0">
    <dict>
      <key>Command</key>
      <dict>
        <key>RequestType</key>
        <string>ScheduleOSUpdate</string>
        <key>Updates</key>
        <array>
          <dict>
            <key>InstallAction</key>
            <string>Default</string>
            <key>ProductKey</key>
            <string>iOSUpdate17A576</string>
            <key>ProductVersion</key>
            <string>13.0</string>
          </dict>
        </array>
      </dict>
      <key>CommandUUID</key>
      <string>0001_ScheduleOSUpdate</string>
    </dict>
  </plist>
```

ScheduleOSUpdate (send)

```
<key>InstallAction</key>  
<string>Default</string>  
<key>ProductKey</key>  
<string>iOSUpdate17A576</string>  
<key>ProductVersion</key>  
<string>13.0</string>
```



System Updates via MDM



DDM Details

- Introduced at WWDC 2021, added to at WWDC 2022
- Nothing necessary from an administration standpoint
- Why is this important?
 - Allows devices to apply management logic without prompting from the server
 - Asynchronous status reporting
 - No need to poll devices
- The way forward for device management

DDM Details

- Platform Support
 - iOS
 - iPadOS
 - macOS
 - tvOS
 - watchOS



DDM Details

- 3 key components
 - Declarations
 - Support device functionality
 - Status
 - Track changes in device state
 - Extensibility
 - Allow devices and servers to communicate the changes in their capabilities



DDM Details

- Declarations
 - 4 types:
 - Configurations
 - Assets
 - Activations
 - Management

DDM Details

- Status
 - Management States
 - Device Properties
 - Other Properties

DDM Details

- Extensibility
 - Allow devices and servers to communicate the changes in their capabilities

Software Updates via DDM

- New functions/features with DDM over MDM
- New status items which can report on:
 - Software update status on a device
 - Details of installation state
 - Failures and failure reasons
- Enhanced user deferral workflow
- DDM software update configurations can coexist with MDM software update commands
- One thing stood out...
 - `softwareupdated` still relevant



Developer

NewsDiscoverDesignDevelopDistributeSupportAccount

Device Management

object ScreenSharingConnectionBeta

object ScreenSharingConnectionDisp...Beta

object ScreenSharingConnectionGroupBeta

object ScreenSharingHostSettingsBeta

object SecurityCertificateBeta

object SecurityIdentityBeta

object SecurityPasskeyAttestationBeta

object ServicesConfigurationFilesBeta

object SoftwareUpdateEnforcemen...Beta

object StatusDeviceModelNumberBeta

object StatusDiskManagementFileVa...Beta

object StatusSecurityCertificateListBeta

object StatusSecurityCertificateListC...Beta

object StatusServicesBackgroundTaskBeta

object StatusServicesBackgroundTas...Beta

object StatusSoftwareUpdateFailureR...Beta

object StatusSoftwareUpdateFailureR...Beta

object StatusSoftwareUpdateInstallR...Beta

object StatusSoftwareUpdateInstallR...Beta

object StatusSoftwareUpdateInstallSt...Beta

object StatusSoftwareUpdatePendin...Beta

object StatusSoftwareUpdatePendin...Beta

Filter

/

Documentation / Device Management / SoftwareUpdateEnforcementSpecificBeta

Language: SwiftAPI Changes: Show

SoftwareUpdateEnforcementSpecificBeta

iOS 17.0+BetaiPadOS 17.0+BetamacOS 14.0+Beta

Properties

DetailsURL	string
TargetBuildVersion	string
TargetLocalDateTime	string (Required)
TargetOSVersion	string (Required)

Beta Software

This documentation contains preliminary information about an API or technology in development. This information is subject to change, and software implemented according to this documentation should be tested with final operating system software.

[Learn more about using Apple's beta software >](#)

Example Configuration

```
com.apple.configuration.softwareupdate.enforcement.specific  
  
{ "TargetOSVersion": "14.0",  
  "TargetBuildVersion": "23A500",  
  "TargetLocalDateTime": "2023-06-05T10:00:00" }
```

TargetOSVersion

OS version the device should update to

TargetBuildVersion

Will take precedence over TargetOSVersion if both are set, if no update is found for the specified version the device will search for the default/recommended update offered by software update service

TargetLocalDateTime

Specifies local date and time when the update will be forced



MDM Watchdog



What is the big deal?

Updates = Security

- `softwareupdated`
- `mdmclient`



Updates Not Installed

Some updates could not be installed.

WHO LET THE DOGS OUT?



How it works!



#1

Checks to see if the device is enrolled into an MDM server

How it works!

#2



Checks to see if the device has a valid identity certificate in place

How it works!

#3



Checks if `mdmclient` process is not stuck using the last 90 minutes of logs from the `mdmclient` process
Part of this is checking the logs for HTTP 200 entries

How it works!

#4



Checks `softwareupdated` using the logs from `mdmclient` and `ManagedClient` subsystem

Looking at the last command received from MDM having to do with the OS Update processes

What then?

MDM Watchdog unloads / reloads services



mdmclient

```
`/bin/launchctl kickstart -k system/com.apple.mdmclient.daemon`
```

softwareupdated

```
`/usr/bin/dscacheutil -flushcache`
```

```
`/usr/bin/killall -HUP mDNSResponder``
```

```
`/usr/bin/pkill BrainService``
```

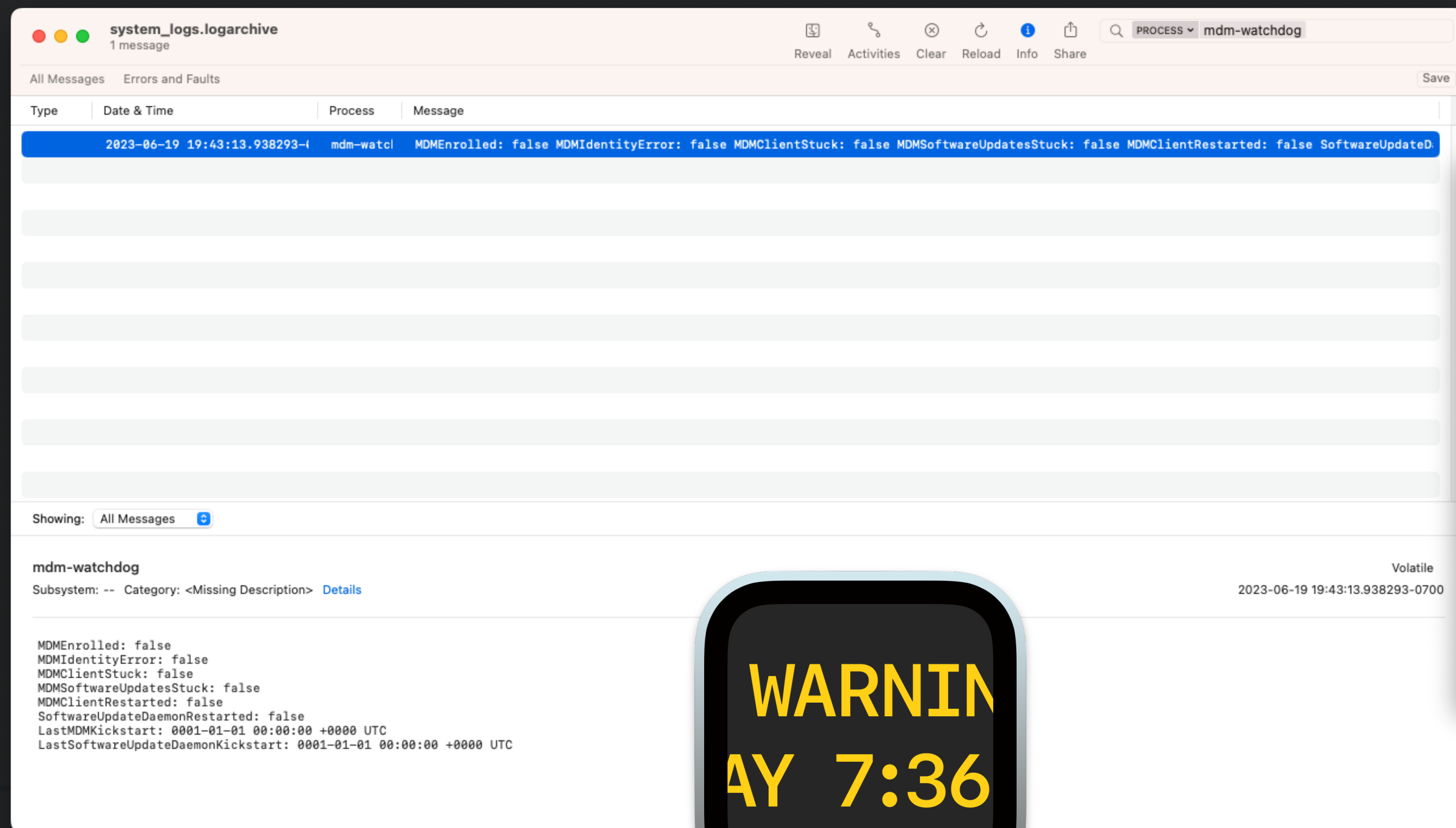
```
`/bin/launchctl kickstart k system/com.apple.softwareupdated`
```

What can I expect?

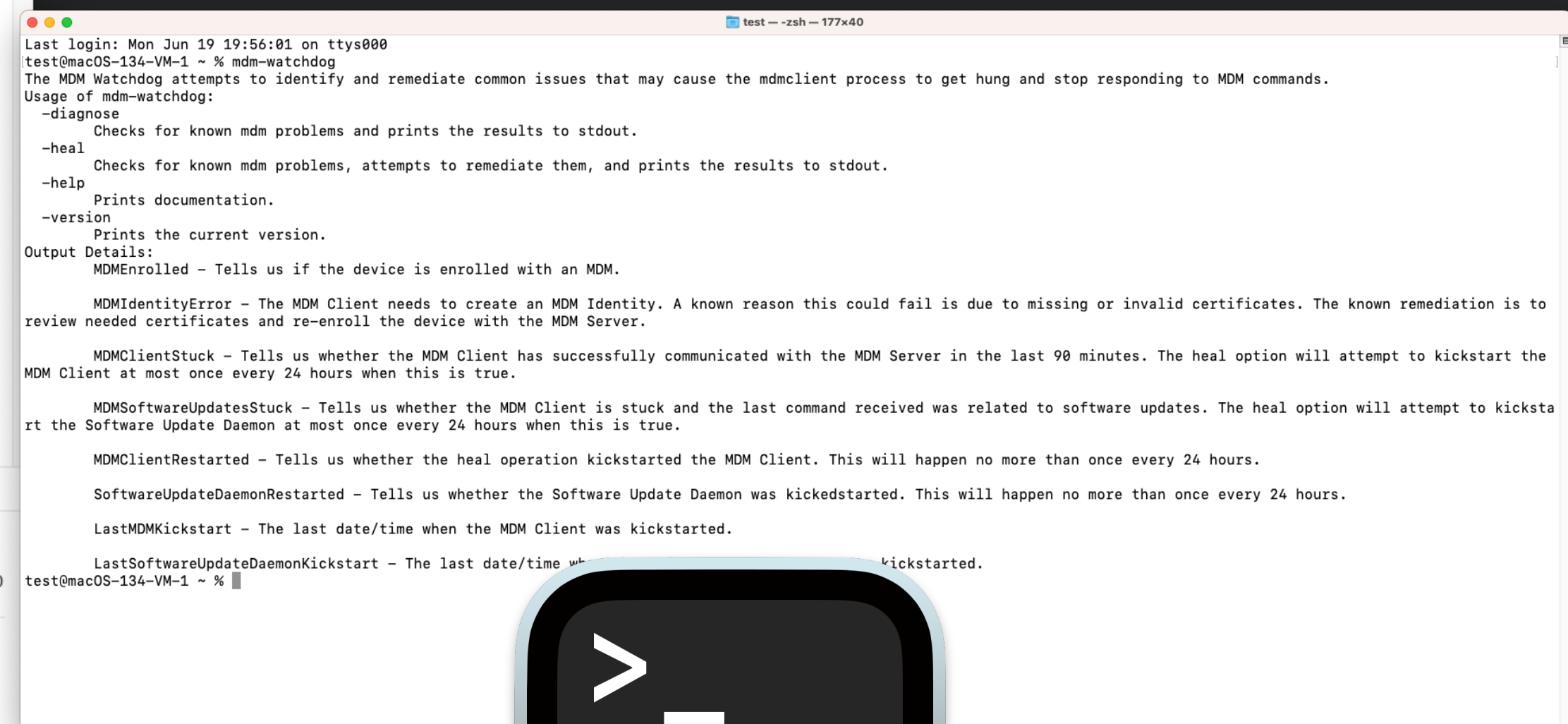
- **/Library/LaunchDaemons**
 - com.addigy.mdm-watchdog.plist
- **Run every 95 minutes**
- **Command Line Tool**
 - mdm-watchdog
- **Does not use network resources or phone home**
 - Does not auto update
 - Updates can be found at <https://addigy.com/mdm-watchdog>

What can I expect?

- log show –predicate ‘process = “mdm-watchdog”’
- Terminal: mdm-watchdog



WARNING
7:36



> _

What can I expect?

- `log show --predicate 'process = "mdm-watchdog"'`

```
default 2023-06-19 19:43:13.938293 -0700 mdm-watchdog MDMEnrolled:
false
MDMIdentityError: false
MDMClientStuck: false
MDMSoftwareUpdatesStuck: false
MDMClientRestarted: false
SoftwareUpdateDaemonRestarted: false
LastMDMKickstart: 0001-01-01 00:00:00 +0000 UTC
LastSoftwareUpdateDaemonKickstart: 0001-01-01 00:00:00 +0000 UTC
```

WARNING
AY 7:36

What can I expect?

•Terminal: `mdm-watchdog`



The MDM Watchdog attempts to identify and remediate common issues that may cause the mdmclient process to get hung and stop responding to MDM commands.

Usage of mdm-watchdog:

-diagnose

Checks for known mdm problems and prints the results to stdout.

-heal

Checks for known mdm problems, attempts to remediate them, and prints the results to stdout.

-help

Prints documentation.

-version

Prints the current version.

Output Details:

MDMEnrolled - Tells us if the device is enrolled with an MDM.

MDMIdentityError - The MDM Client needs to create an MDM Identity. A known reason this could fail is due to missing or invalid certificates. The known remediation is to review needed certificates and re-enroll the device with the MDM Server.

MDMClientStuck - Tells us whether the MDM Client has successfully communicated with the MDM Server in the last 90 minutes. The heal option will attempt to kickstart the MDM Client at most once every 24 hours when this is true.

MDMSoftwareUpdatesStuck - Tells us whether the MDM Client is stuck and the last command received was related to software updates. The heal option will attempt to kickstart the Software Update Daemon at most once every 24 hours when this is true.

MDMClientRestarted - Tells us whether the heal operation kickstarted the MDM Client. This will happen no more than once every 24 hours.

SoftwareUpdateDaemonRestarted - Tells us whether the Software Update Daemon was kickedstarted. This will happen no more than once every 24 hours.

LastMDMKickstart - The last date/time when the MDM Client was kickstarted.

LastSoftwareUpdateDaemonKickstart - The last date/time when the Software Update Daemon was kickstarted.

What can I expect?

- **Terminal:** `mdm-watchdog -diagnose`

```
bryce@macOS-134-VM-1 ~ % mdm-watchdog -diagnose
MDMEnrolled: false
MDMIdentityError: false
MDMClientStuck: false
MDMSoftwareUpdatesStuck: false
```


MDM Watchdog in Addigy vs Standalone with any MDM

	Addigy	Other MDMs
Run Frequency	Every 5 min. audit then 30 min. agent	Every 95 minutes
Reporting	GoLive, Alert, and Dashboard	log show or scripted CMD
Identity Certificate	Built in device fact	log show or scripted CMD
MDM Client Stuck	Built in device fact	log show or scripted CMD
Software Update Stuck	Built in device fact	log show or scripted CMD
Software Updates via MDM Retry Logic	24 hour default; 1 to 24 hour increments	Depends
Deployment	Part of Addigy binary	.pkg
Updates	Part of Addigy binary	New .pkg

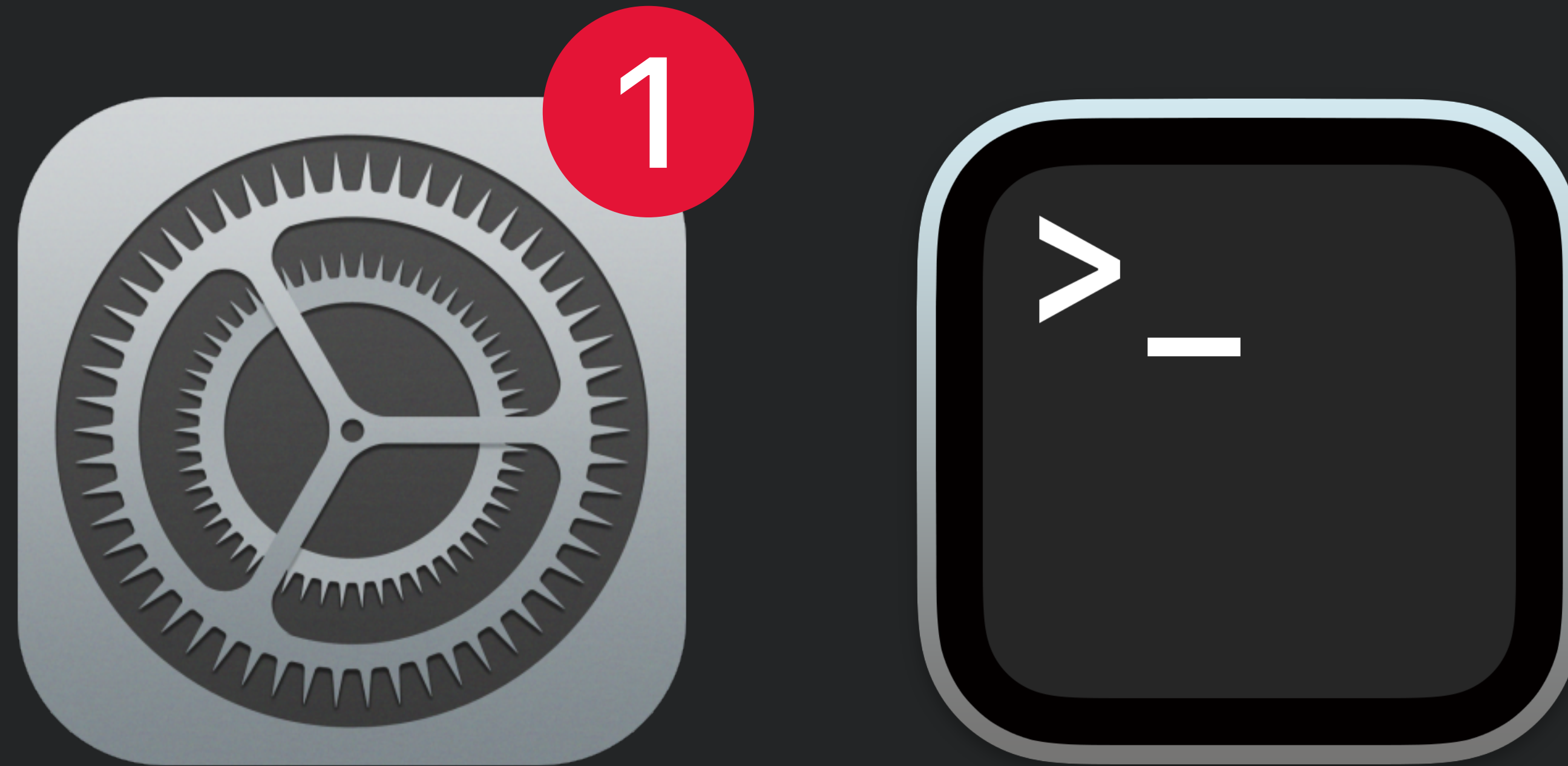
MDM Watchdog



- <https://addigy.com/mdm-watchdog/>

- <https://agents.addigy.com/tools/mdm-watchdog/latest/mdm-watchdog.pkg>

System Updates via Script/Public Software



System Updates The Other Alternative

Catalog

Saved and reusable items for your policies. Any changes to an item will affect all uses.

Files

MDM Profiles

Software

Monitoring

Maintenance

Custom Facts

Self Service

OS Users

Compliance

Smart Software

Public Library

Manage software deployments from the public library to your policy's devices.

ventura

Addigy-supported software only

Actions	Name	Uploaded by	Org. Usage	Condition Script	Removal Script ↑
...	Install macOS Ventura (13.0.0.1)	Addigy	11	●	●
...	Install macOS Ventura (13.0.1.1)	Addigy	12	●	●
...	Install macOS Ventura (13.1.0.1)	Addigy	6	●	●
...	Install macOS Ventura (13.2.0.1)	Addigy	9	●	●
...	Install macOS Ventura (13.2.1.1)	Addigy	10	●	●
...	Install macOS Ventura (13.3.0.1)	Addigy	6	●	●
...	Install macOS Ventura (13.3.1.1)	Addigy	28	●	●
...	Install macOS Ventura (13.4.0.1)	Addigy	58	●	●

System Updates The Other Alternative

Auto-Assignment

Policy: macOS Patching Ventura

☒ Enabled

Add filters to assign devices automatically

Build a filter set to test devices when they check in. Any device that meets the set of criteria will receive any settings and assets included in this policy.

MAC OS X Version	wildcard	11*	×
macOS Ventura Support	=	<input checked="" type="checkbox"/>	×

 Add filter

OR

 Add filter

Test Filters

☐ Unassign devices that no longer match this filter set
Manually assigned devices will also be removed.

Delete Auto-Assignment


Note: Devices will no longer be assigned automatically. Any devices that were assigned to this policy can be removed manually.

 Delete

Close

Save auto-assignment

Software Update User Experience




Install macOS Ventura

An upgrade is required by your IT Team and will require a reboot.

After clicking 'Install', your device will start the upgrade process. The device will reboot in about 10-20 minutes after the upgrade starts.


Not NowInstall



Your IT Admin requires your password to continue the macOS Upgrade.

Enter the password for user "admin" to continue the upgrade:

Continue



Upgrade in Progress


This upgrade requires a reboot.
Estimated time of reboot:
11:41 PM - 11:51 PM

Please save all work.

Software Updates & Addigy LANCache

We understand that downloading a full macOS installer for patching might not be ideal, but Addigy has a solution with LANCache.

"The LANCache utility is automatically deployed with all Addigy agents and helps distribute files and packages across the local area network (LAN). This helps reduce the amount of internet bandwidth used when downloading software to devices on the same LAN."



Questions?

Also, come visit us

1-1 Questions & Demos

Giveaway - Apple AirPods Pro

addigy.com

@addigy



MacAdmins #addigy



Thanks!

addigy.com

@addigy



MacAdmins #addigy