MDM

Charles Edge

Agenda

- History
- Types of Profiles
- The MDM Check-In Protocol
- The MDM Protocol
- VPP
- Best Practices

A Brief History Of Time

2008

Israel invades the Gaza Strip

North Korea Claims Denuclearization

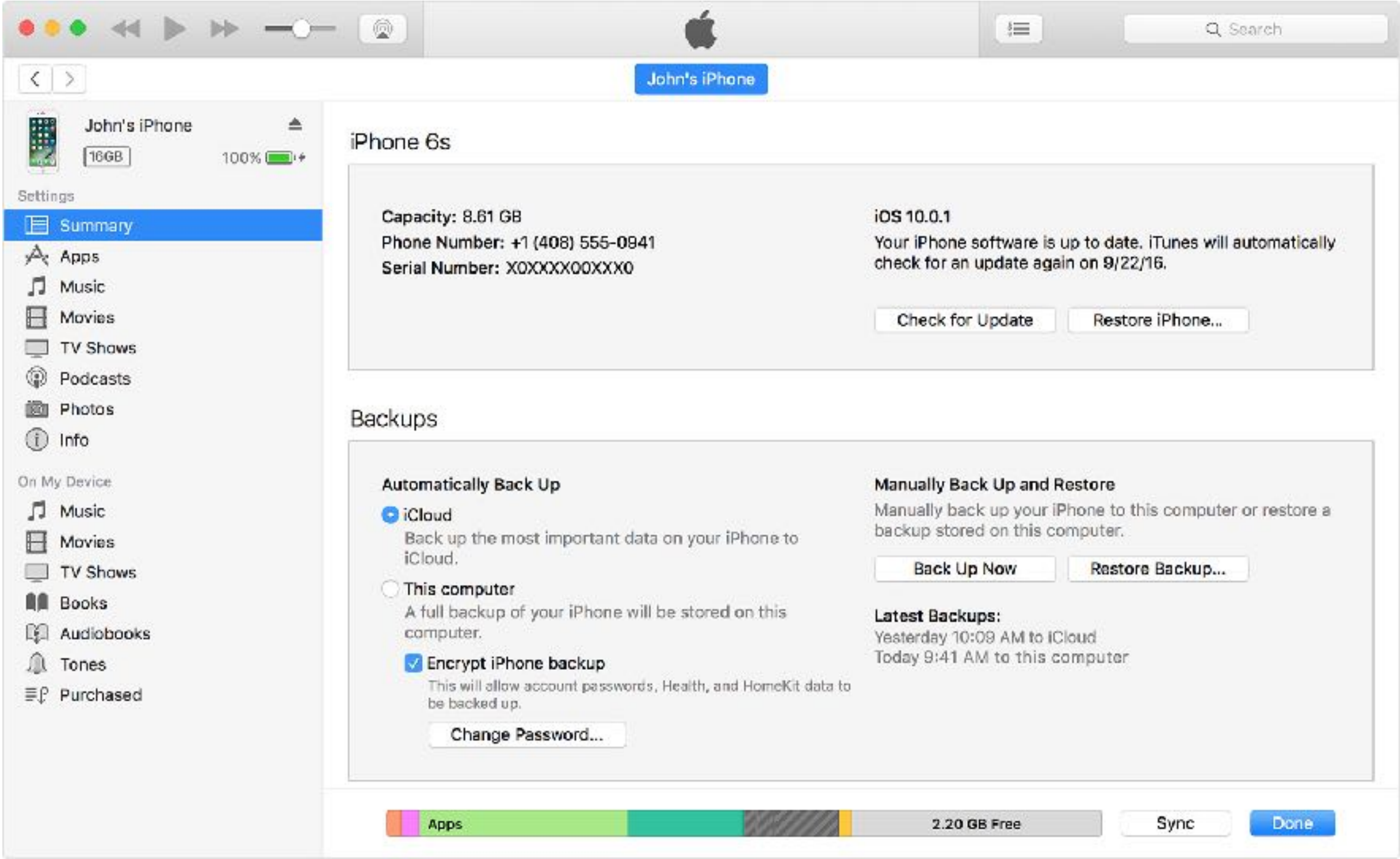
Robert Mugabe Re-elected in Zimbabwe

Hillary Clinton threatens to “obliterate” Iran

iPhone OS 2

Introduction of EAS support and Configuration Profiles

My first big iOS deployment



iPhone 6s

Capacity: 8.61 GB
Phone Number: +1 (408) 555-0941
Serial Number: X0XXXX00XXX0

iOS 10.0.1
Your iPhone software is up to date. iTunes will automatically check for an update again on 9/22/16.

Backups

Automatically Back Up

● iCloud
Back up the most important data on your iPhone to iCloud.

○ This computer
A full backup of your iPhone will be stored on this computer.

☑ Encrypt iPhone backup
This will allow account passwords, Health, and HomeKit data to be backed up.

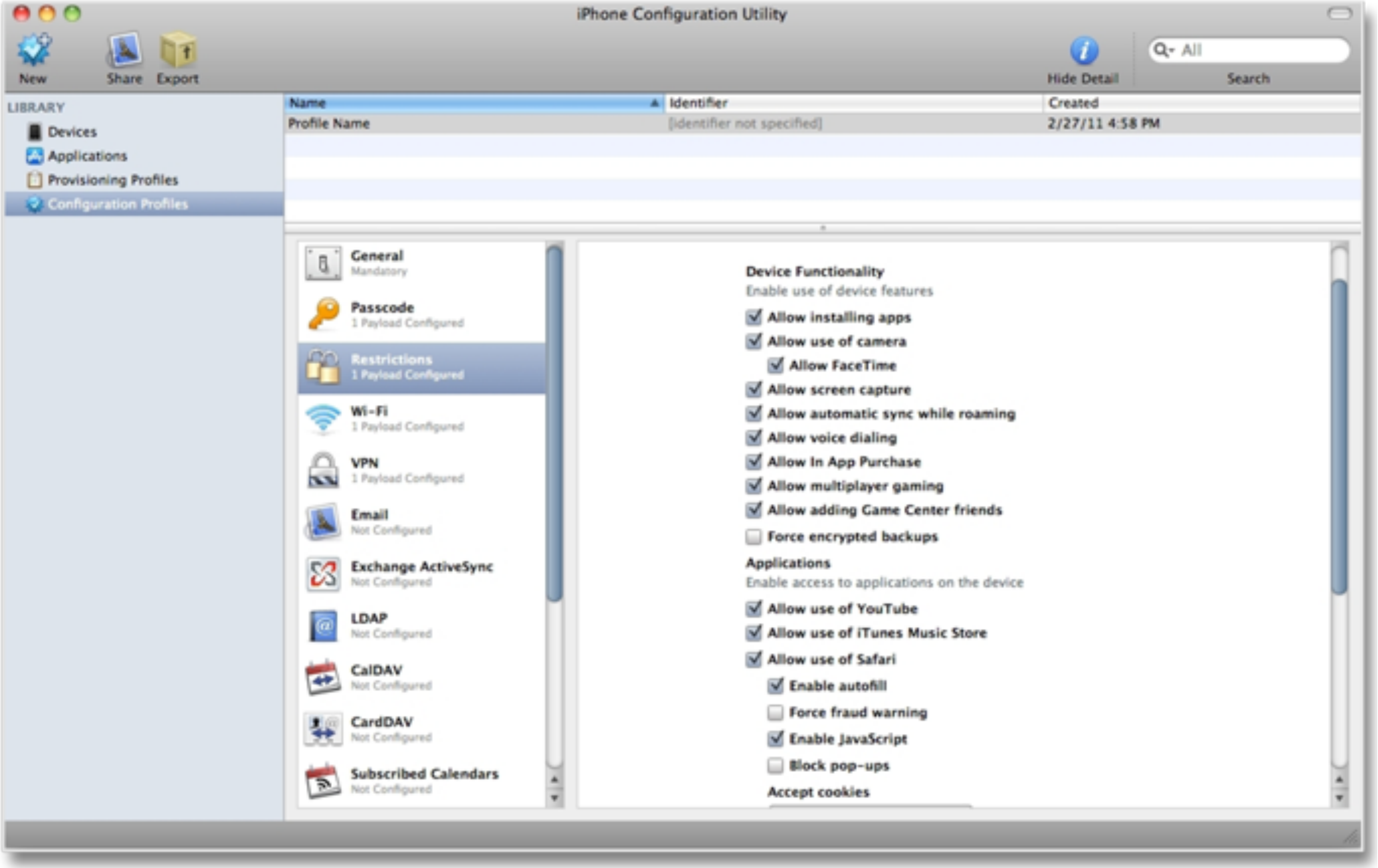
Manually Back Up and Restore
Manually back up your iPhone to this computer or restore a backup stored on this computer.

Latest Backups:
Yesterday 10:09 AM to iCloud
Today 9:41 AM to this computer

Hotmail

The 1st Gen Of Management Tools

iPhone Configuration Utility



Name	Identifier	Created
Profile Name	[identifier not specified]	2/27/11 4:58 PM

Device Functionality

Enable use of device features

☑ Allow installing apps

☑ Allow use of camera

☑ Allow FaceTime

☑ Allow screen capture

☑ Allow automatic sync while roaming

☑ Allow voice dialing

☑ Allow In App Purchase

☑ Allow multiplayer gaming

☑ Allow adding Game Center friends

□ Force encrypted backups

Applications

Enable access to applications on the device

☑ Allow use of YouTube

☑ Allow use of iTunes Music Store

☑ Allow use of Safari

☑ Enable autofill

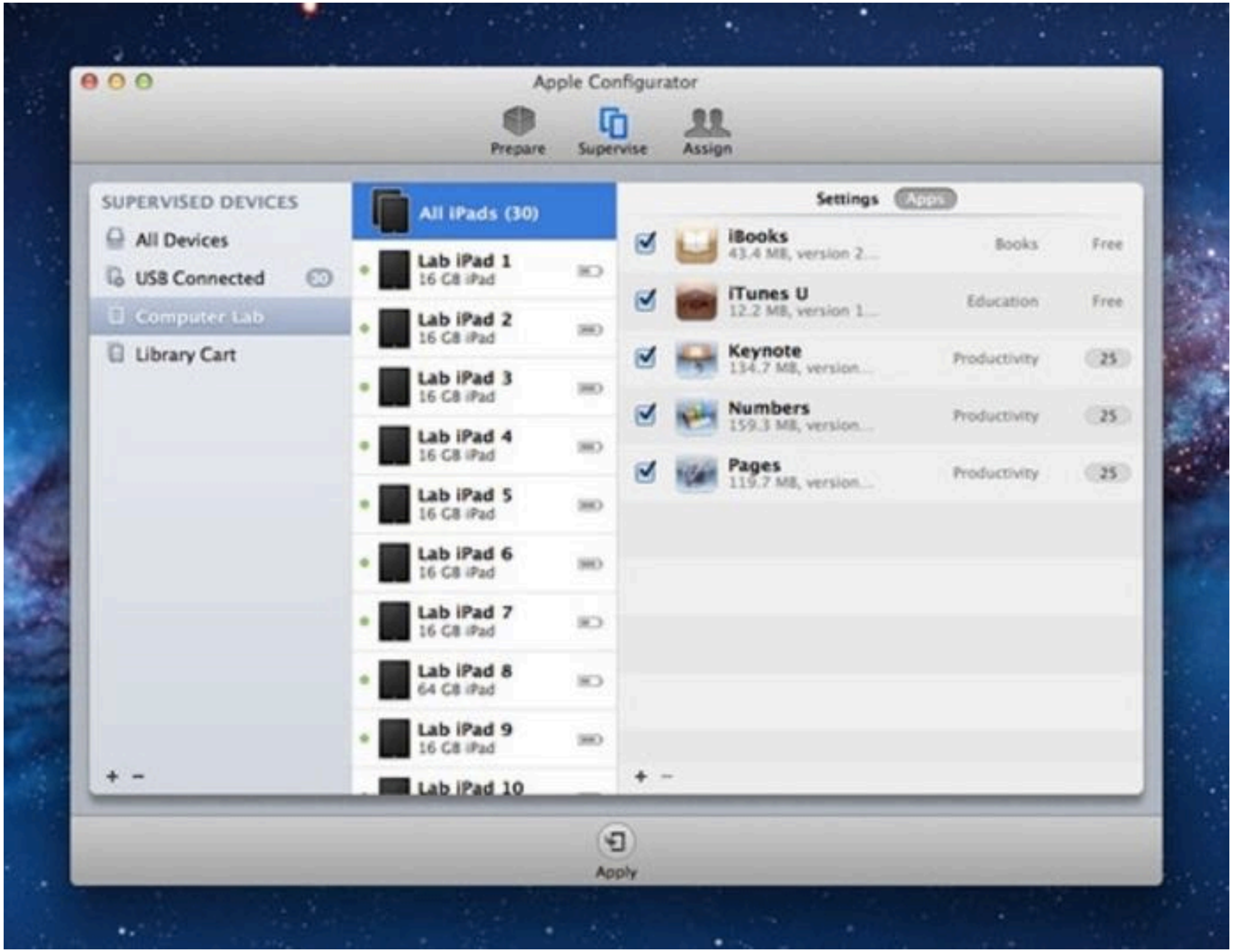
□ Force fraud warning

☑ Enable JavaScript

□ Block pop-ups

Accept cookies

Apple Configurator



All iPads (30)

Lab iPad 1
16 GB iPad

Lab iPad 2
16 GB iPad

Lab iPad 3
16 GB iPad

Lab iPad 4
16 GB iPad

Lab iPad 5
16 GB iPad

Lab iPad 6
16 GB iPad

Lab iPad 7
16 GB iPad

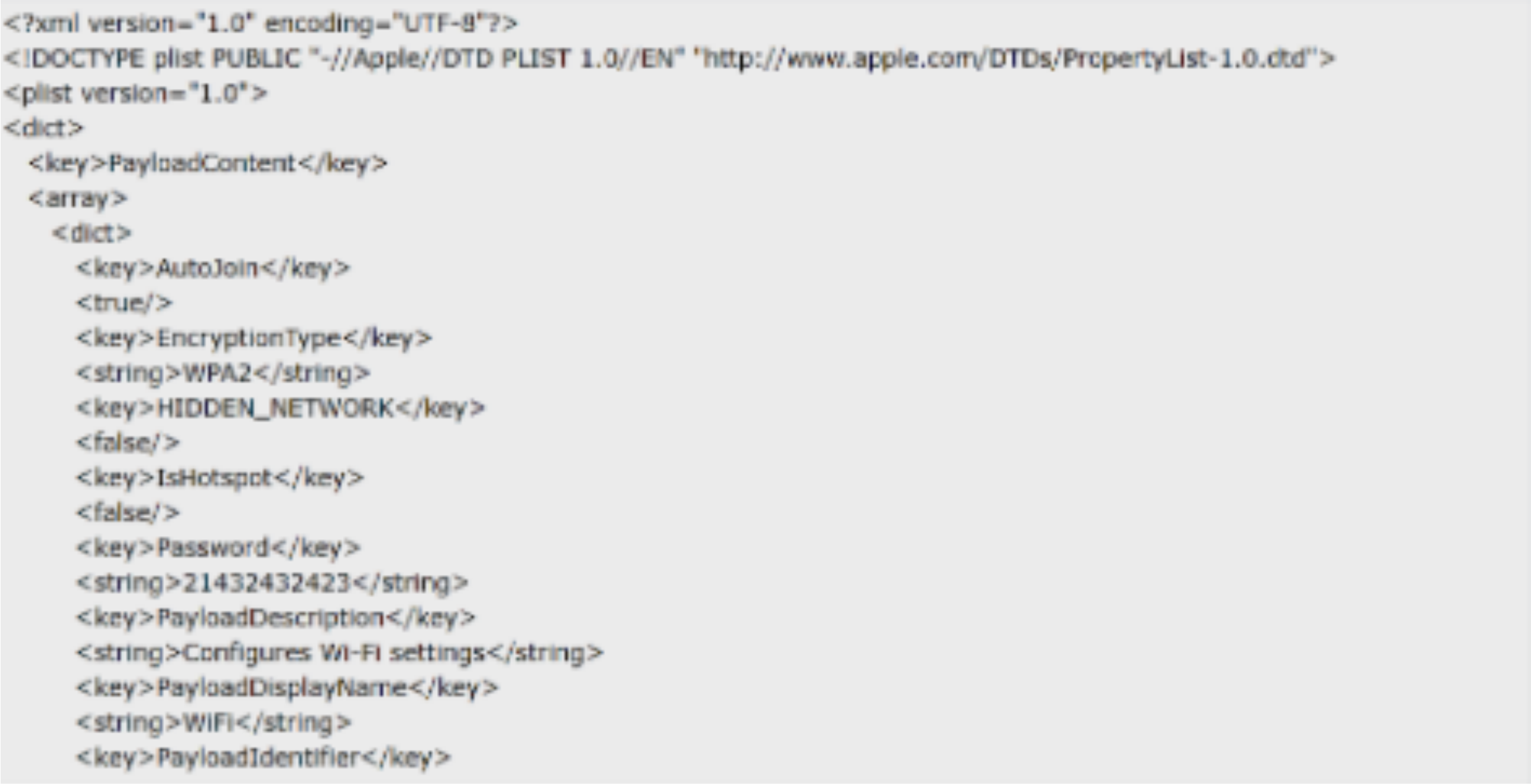
Lab iPad 8
64 GB iPad

Lab iPad 9
16 GB iPad

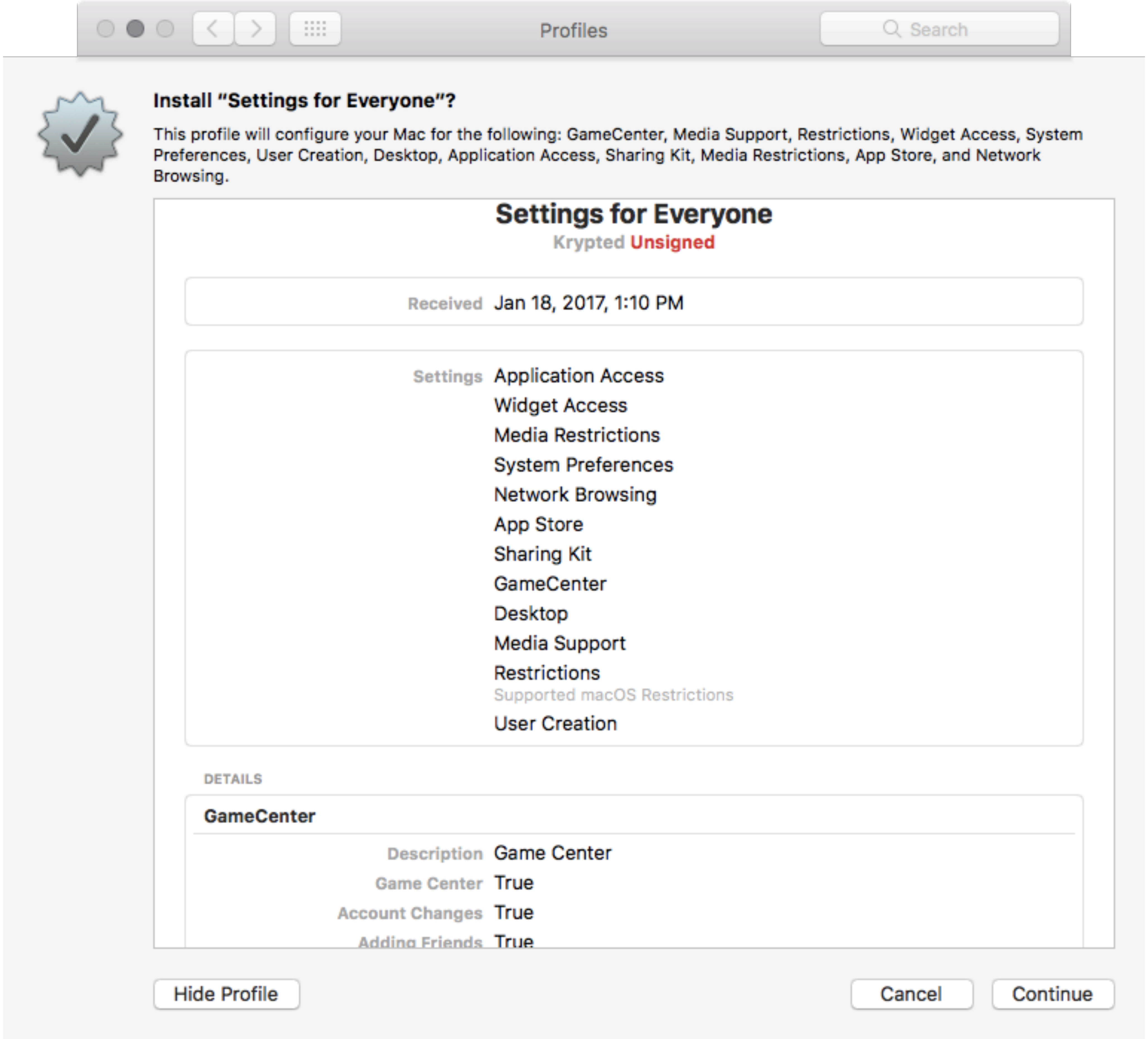
Lab iPad 10

☑	iBooks 43.4 MB, version 2…	Books	Free
☑	iTunes U 12.2 MB, version 1…	Education	Free
☑	Keynote 134.7 MB, version…	Productivity	25
☑	Numbers 159.3 MB, version…	Productivity	25
☑	Pages 119.7 MB, version…	Productivity	25

Profiles



```
<?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>PayloadContent</key>
  <array>
    <dict>
      <key>AutoJoin</key>
      <true/>
      <key>EncryptionType</key>
      <string>WPA2</string>
      <key>HIDDEN_NETWORK</key>
      <false/>
      <key>IsHotspot</key>
      <false/>
      <key>Password</key>
      <string>21432432423</string>
      <key>PayloadDescription</key>
      <string>Configures Wi-Fi settings</string>
      <key>PayloadDisplayName</key>
      <string>WiFi</string>
      <key>PayloadIdentifier</key>
```



Install "Settings for Everyone"?

This profile will configure your Mac for the following: GameCenter, Media Support, Restrictions, Widget Access, System Preferences, User Creation, Desktop, Application Access, Sharing Kit, Media Restrictions, App Store, and Network Browsing.

Settings for Everyone

Krypted **Unsigned**

Received	Jan 18, 2017, 1:10 PM

Settings	Application Access Widget Access Media Restrictions System Preferences Network Browsing App Store Sharing Kit GameCenter Desktop Media Support Restrictions Supported macOS Restrictions User Creation

DETAILS

GameCenter

Description	Game Center
Game Center	True
Account Changes	True
Adding Friends	True

Can be created programmatically (e.g. mcxToProfile.py)

https://github.com/timsutton/mcxToProfile/blob/master/mcxToProfile.py

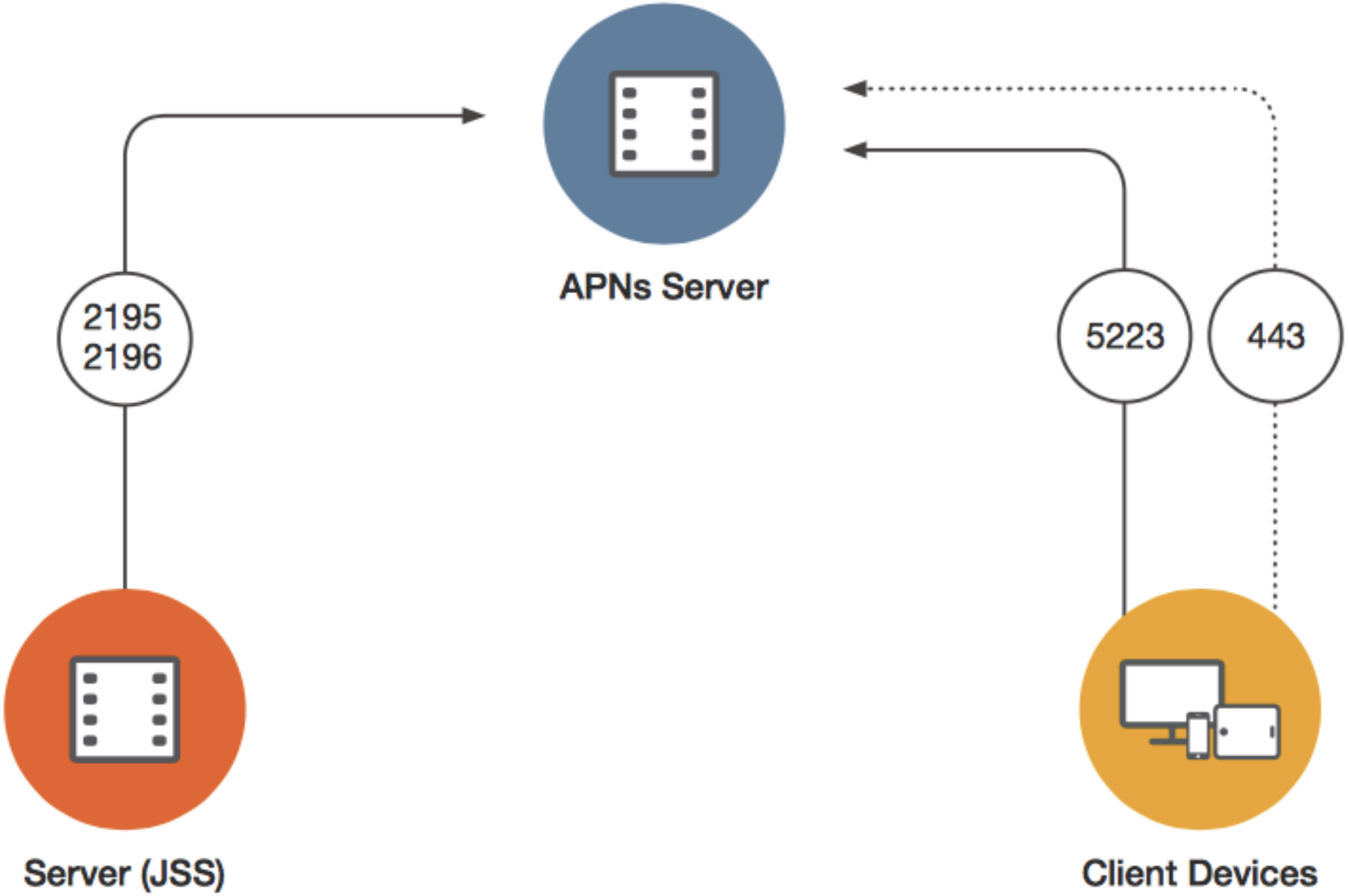
Can be managed manually

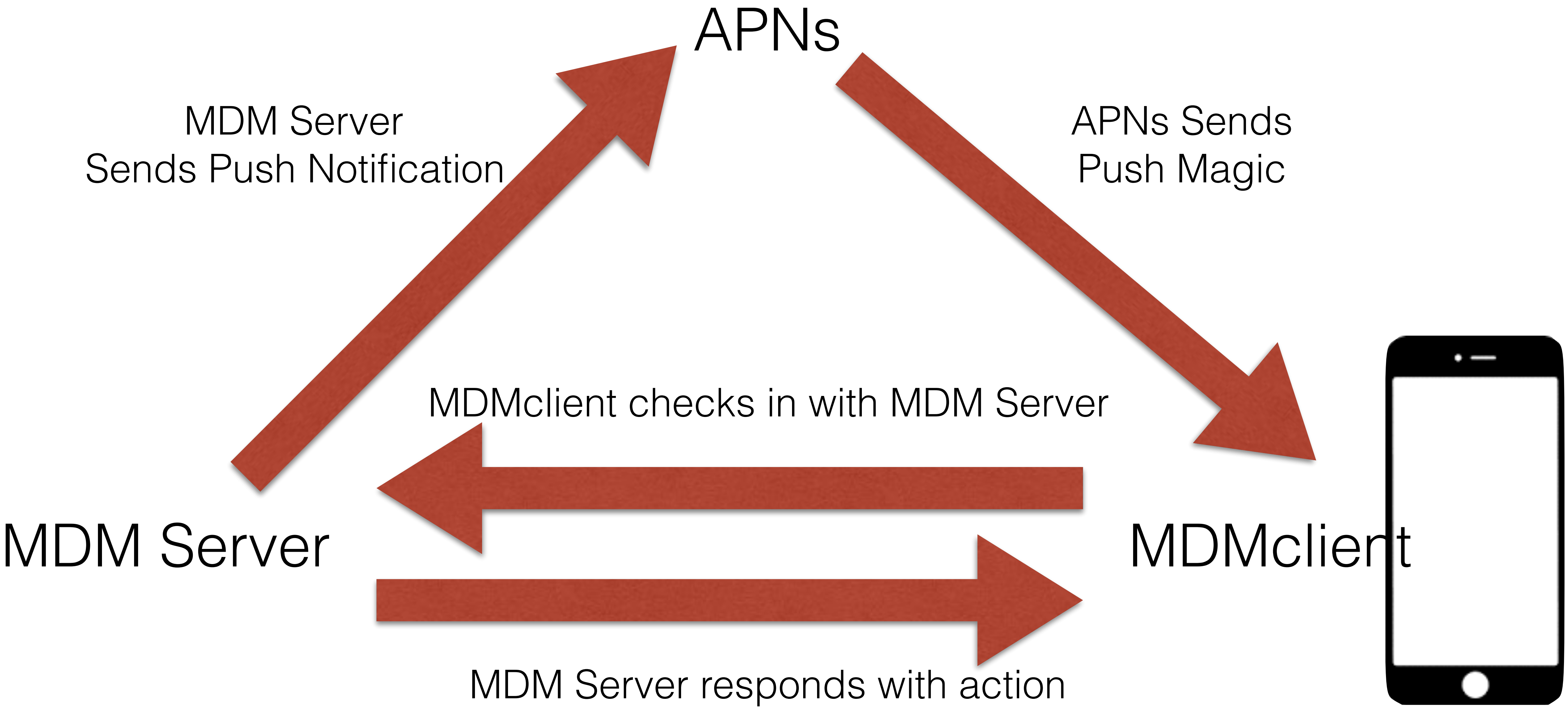
http://krypted.com/mac-security/manage-profiles-from-the-command-line-in-os-x-10-9/

Management companies built profile installers

All management was opt-in

Then came MDM


APNs Server
2195
2196
5223
443
Server (JSS)
Client Devices



APNs

MDM Server
Sends Push Notification

APNs Sends
Push Magic

MDMclient checks in with MDM Server

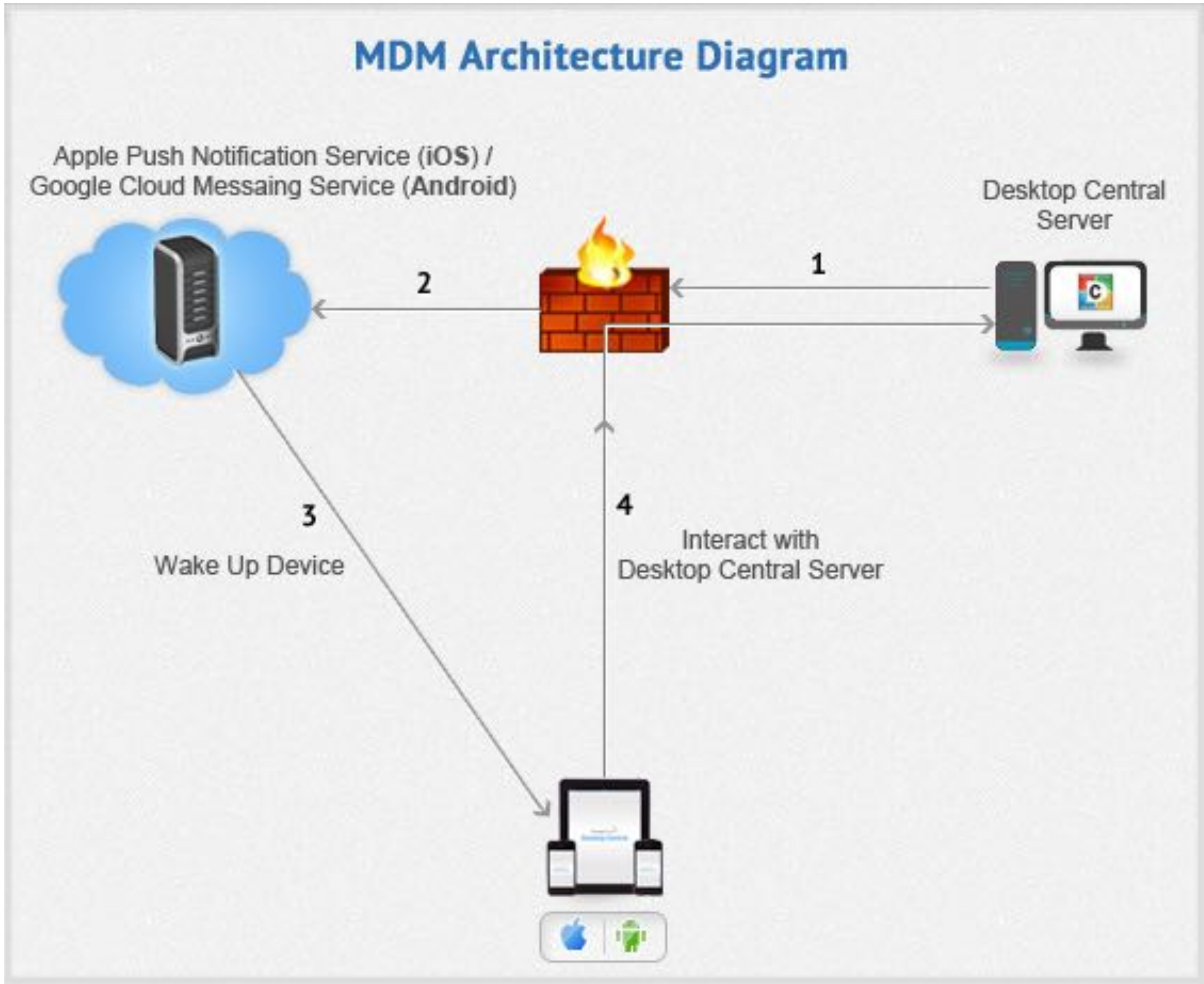
MDM Server

MDMclient

MDM Server responds with action

And it works, so Google borrowed it

Image from ManageEngine

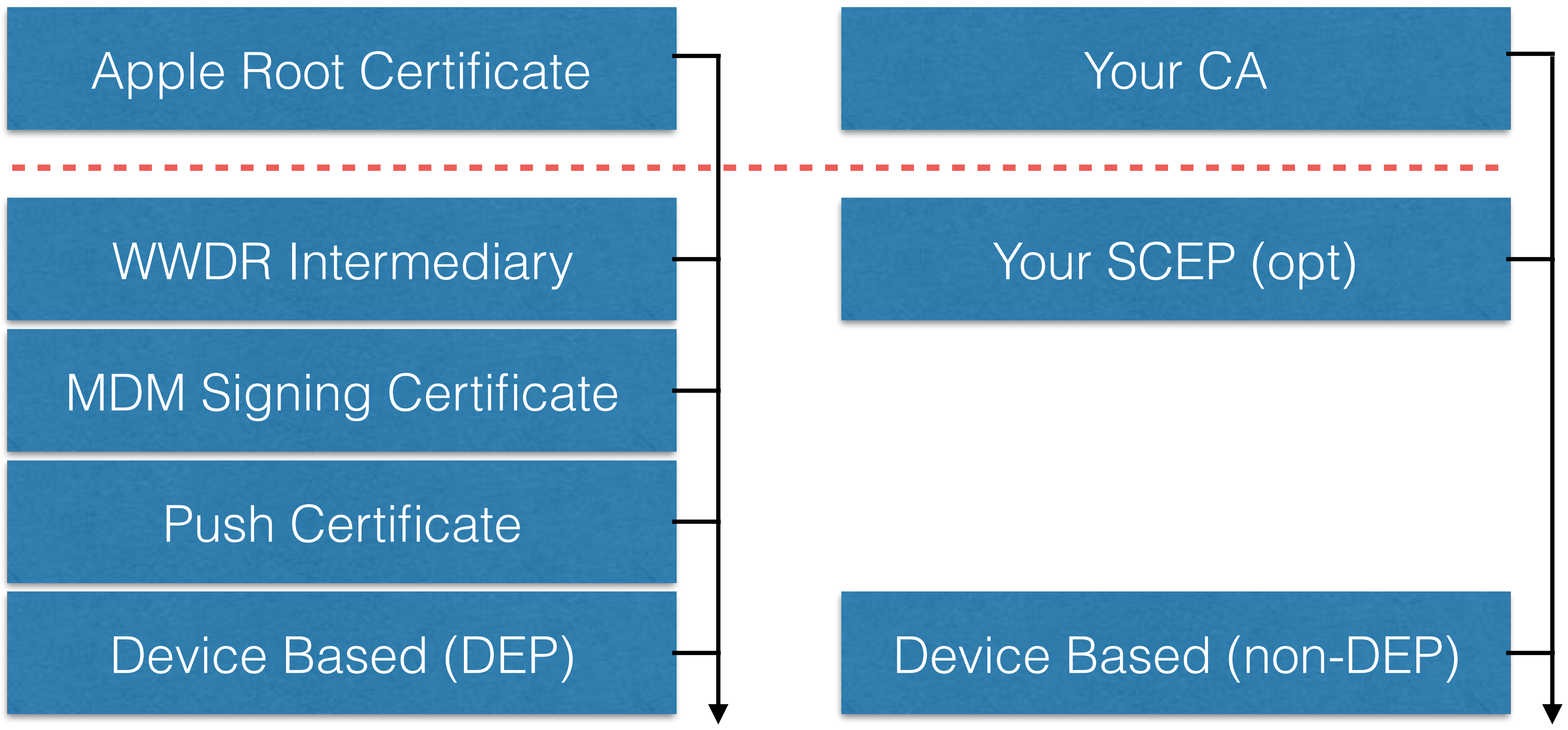


The MDM Spec

> “It’s always the certificates that are a pain”
>
> A Developer, Monday the 26th

Why Are Certificates A Pain?

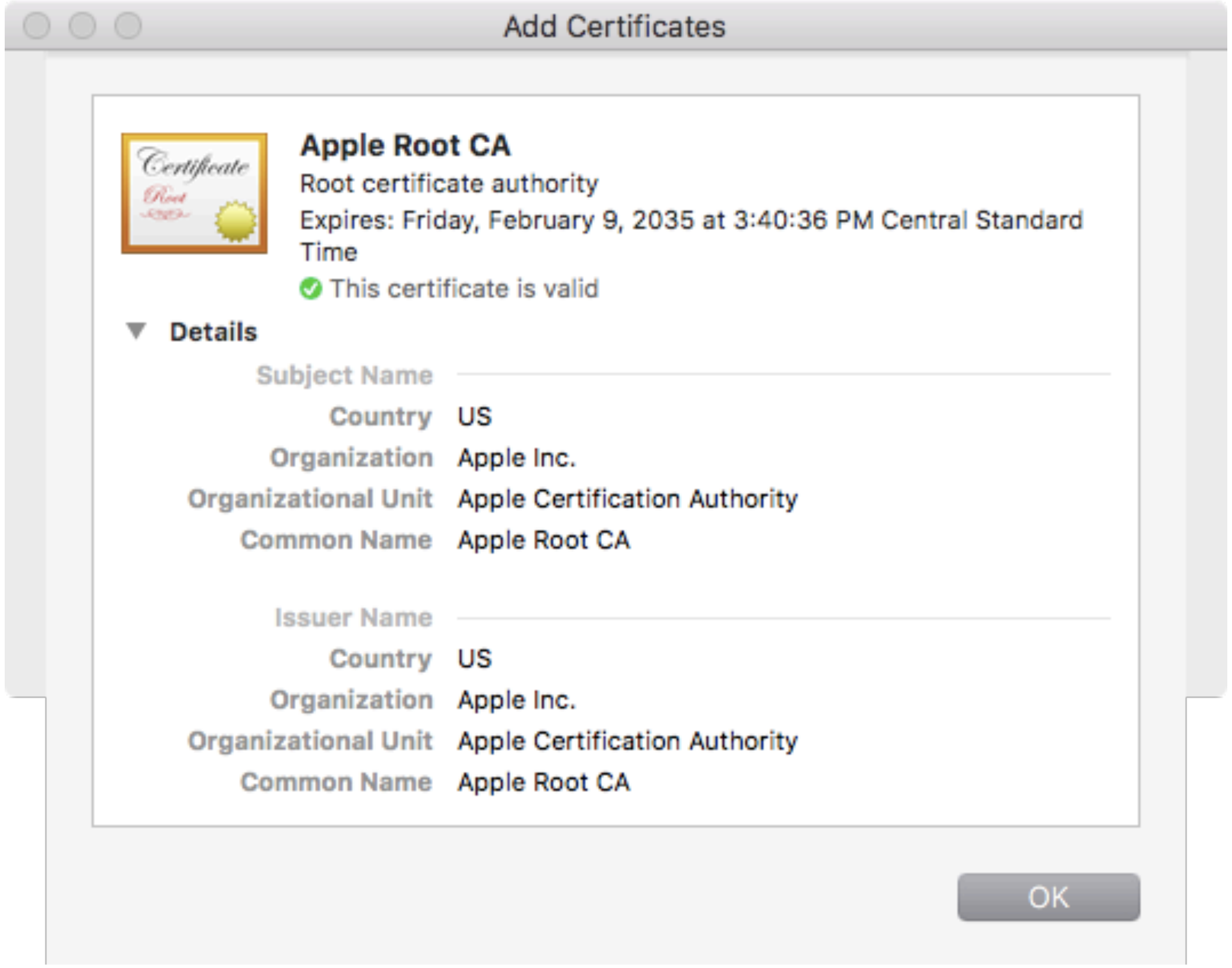
The Certificate Chain



The Beginning of the Certificate Chain

- WWDR intermediate certificate: http://developer.apple.com/certificationauthority/AppleWWDRCA.cer
- Apple root certificate: http://www.apple.com/appleca/AppleIncRootCertificate.cer

Apple Root Certificate



Apple Root CA

Root certificate authority

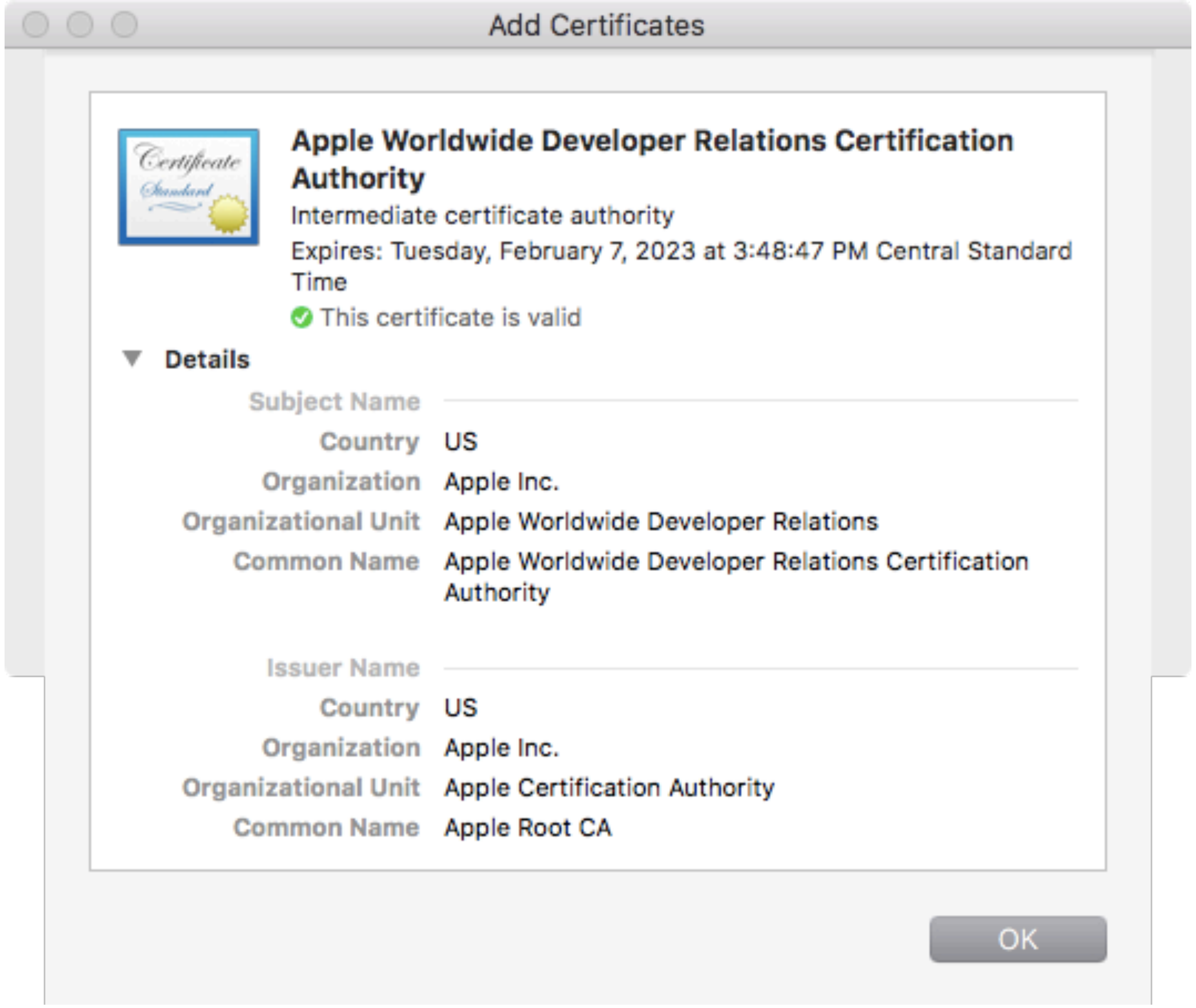
Expires: Friday, February 9, 2035 at 3:40:36 PM Central Standard Time

✓ This certificate is valid

▼ **Details**

Subject Name	
Country	US
Organization	Apple Inc.
Organizational Unit	Apple Certification Authority
Common Name	Apple Root CA
Issuer Name	
Country	US
Organization	Apple Inc.
Organizational Unit	Apple Certification Authority
Common Name	Apple Root CA

WWDR Intermediary



Apple Worldwide Developer Relations Certification Authority

Intermediate certificate authority

Expires: Tuesday, February 7, 2023 at 3:48:47 PM Central Standard Time

✓ This certificate is valid

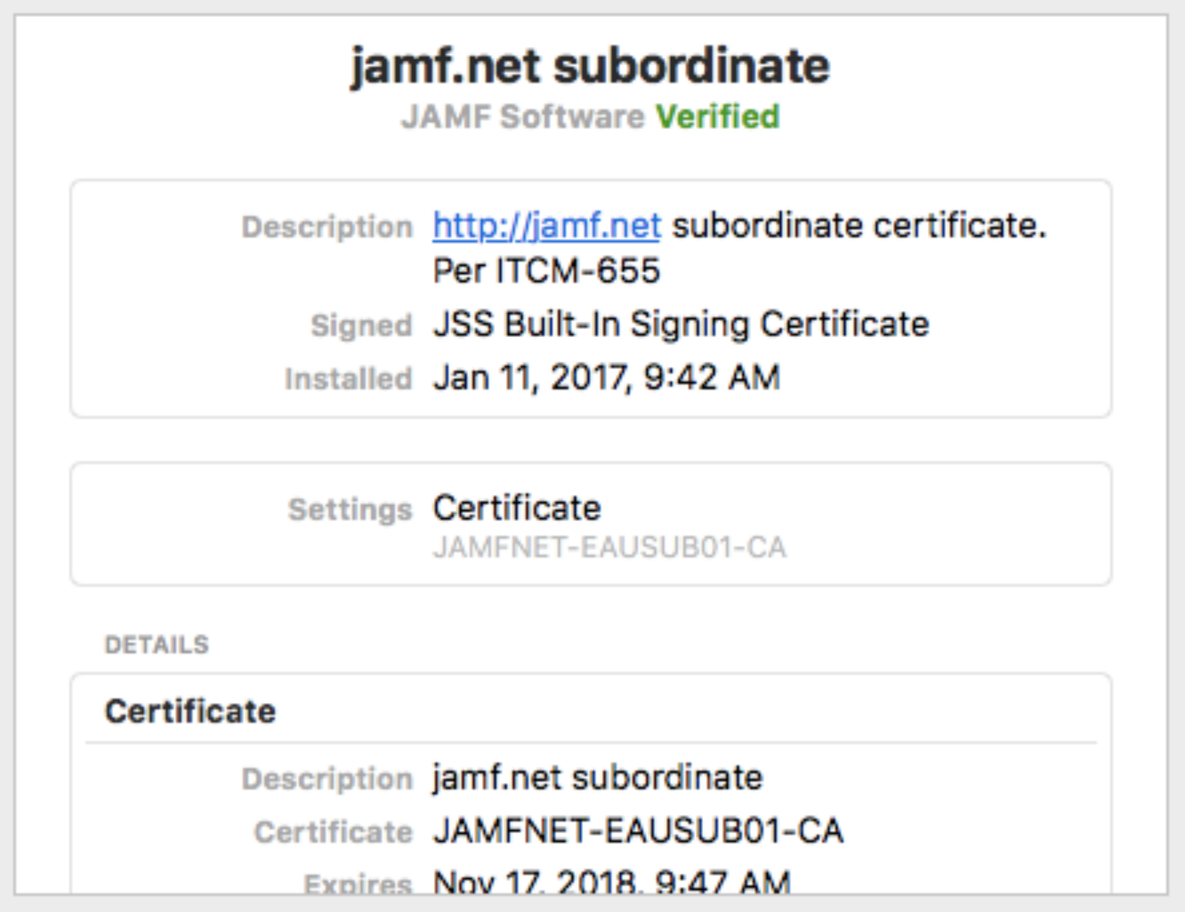
▼ **Details**

Subject Name	
Country	US
Organization	Apple Inc.
Organizational Unit	Apple Worldwide Developer Relations
Common Name	Apple Worldwide Developer Relations Certification Authority
Issuer Name	
Country	US
Organization	Apple Inc.
Organizational Unit	Apple Certification Authority
Common Name	Apple Root CA

MDM Signing Certificate

MDM Signing Certificate

- Establishes trust between MDM vendor/provider and Apple to be able to do APNs
- Obtained from the iOS Provisioning portal so was restricted to vendors
- Contains a private key, public keys and trust certificates
- Used to sign a customer's CSR
- As with all private keys, the private key should stay private
- Expire



jamf.net subordinate

JAMF Software Verified

Description http://jamf.net subordinate certificate.
Per ITCM-655

Signed JSS Built-In Signing Certificate

Installed Jan 11, 2017, 9:42 AM

Settings Certificate
JAMFNET-EAUSUB01-CA

DETAILS

Certificate

Description jamf.net subordinate

Certificate JAMFNET-EAUSUB01-CA

Expires Nov 17, 2018, 9:47 AM

Certificate Signing Request (CSR)

CSR

- Must be in DER (binary)
- Signed w/ the private key of the MDM Signing Cert
- Signed with SHA1WithRSA
- Signature and CSR are base64 encoded
- Push Certificate Request is generated as a base64 plist

CSR (cont)

- PushCertWebRequest is a file downloaded by admins
- File is uploaded to https://identity.apple.com/pushcert
- Certificate is downloaded as MDM_<VendorName>_Certificate.pem and uploaded to the MDM solution
- MDM Solution can then do Apple Push Notifications

Device Identity Certificate

Device Identity Certificate

- Used to encrypt profiles sent to devices
- Any time a device checks in, validate that the certificate was signed against the CA as the device includes the certificate at each checkin
- DEP devices bootstrap with a certificate signed by Apple

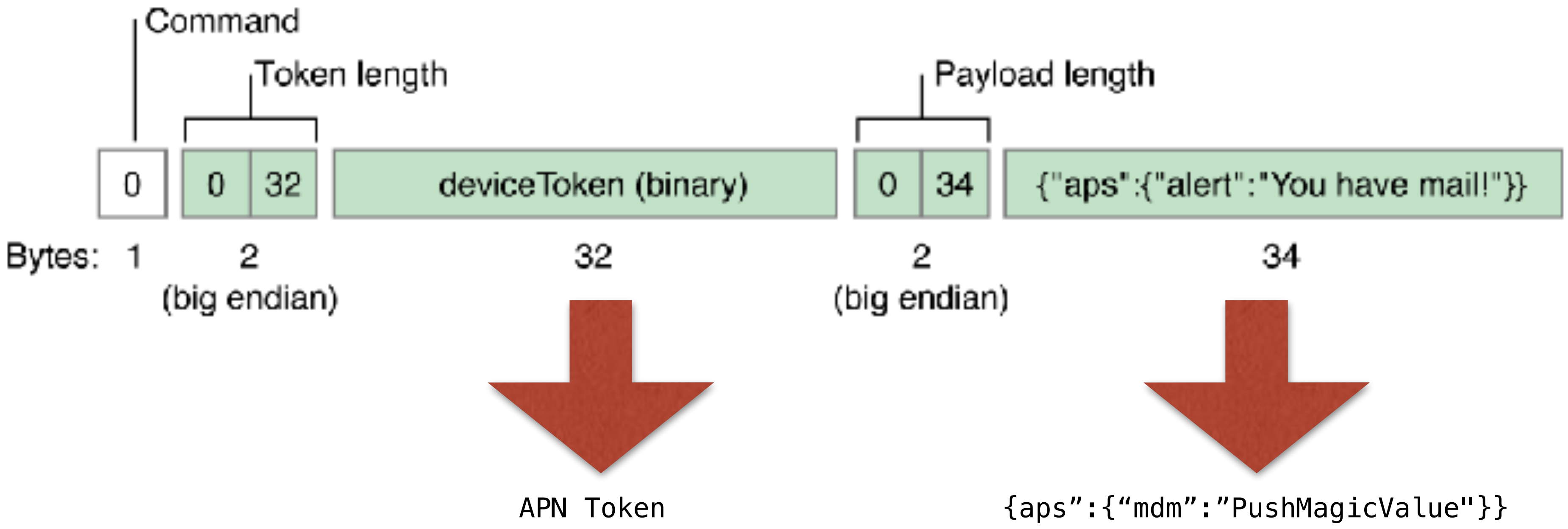
APNs Token (aka Device Token)

- String broken up, each is sent in push notifications in binary
- Stored as 32 binary characters

With All These Certificates, Wat Could Go Wrong?!?!

Why Do I Have To Open Port 2195?

Glue It Together



Command

Token length

Payload length

0	0	32	deviceToken (binary)	0	34	{"aps":{"alert":"You have mail!"}}

Bytes: 1

2 (big endian)

32

2 (big endian)

34

APN Token

{aps”:{“mdm”:”PushMagicValue"}}

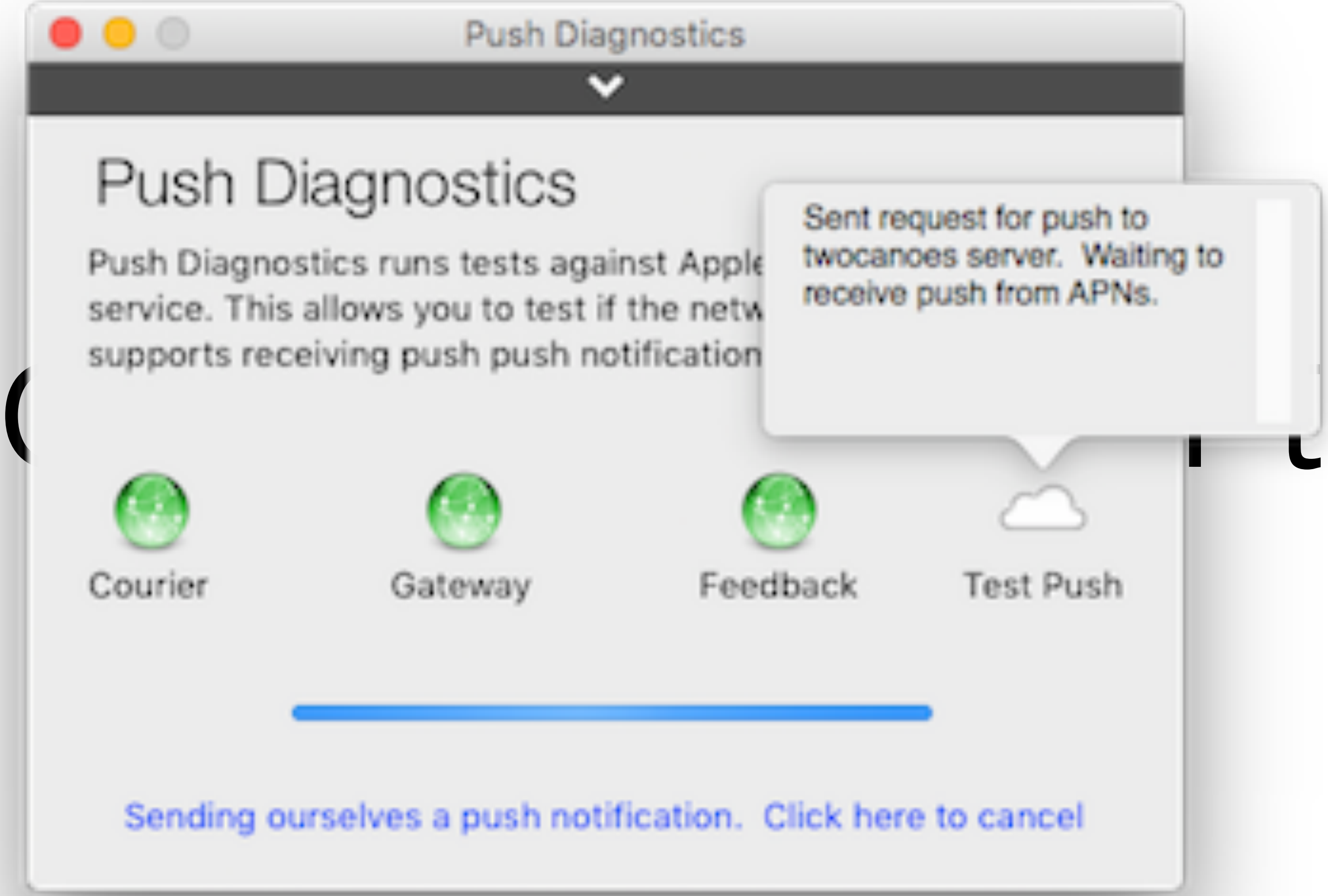
gateway.push.apple.com:2195

gateway.push.apple.com:443

For More On APNs

https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/CommunicatingwithAPNs.html#//apple_ref/doc/uid/TP40008194-CH11-SW1

If y the



Push Diagnostics

Push Diagnostics runs tests against Apple
service. This allows you to test if the netw
supports receiving push push notification

Sent request for push to twocanoes server. Waiting to receive push from APNs.

Sending ourselves a push notification. Click here to cancel

Test It

```
telnet gateway.push.apple.com 2195
```

This is outgoing traffic

What IP range again?

17.0.0.0/8

Feedback (port 2196) checks if devices still have tokens

Devices Talk Back Over 5223

Can fall back to 443
over wi-fi

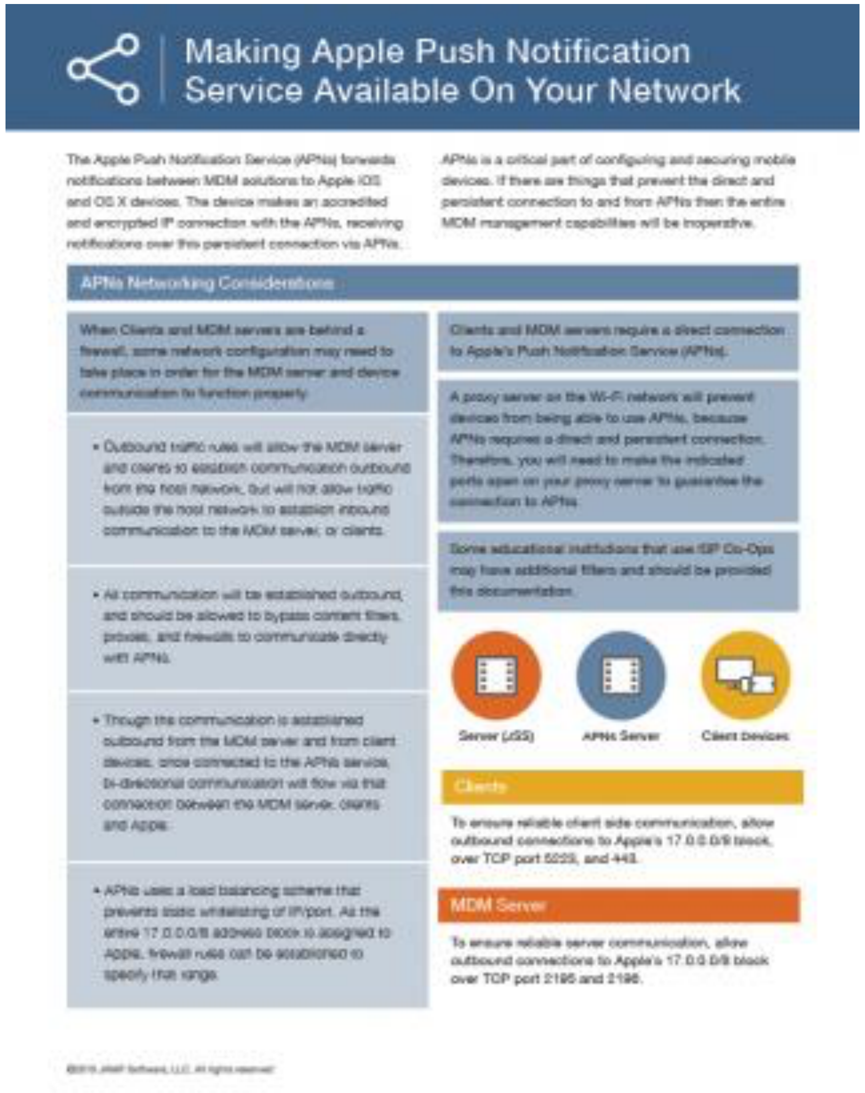
```
telnet 1-courier.push.apple.com 5223
```

A 410 error means the device token is expired

Moar Troubleshooting

https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/CommunicatingwithAPNs.html#//apple_ref/doc/uid/TP40008194-CH11-SW1

Basic stuffs



Making Apple Push Notification Service Available On Your Network

The Apple Push Notification Service (APNs) forwards notifications between MDM solutions to Apple iOS and OS X devices. The device makes an accredited and encrypted IP connection with the APNs, receiving notifications over this persistent connection via APNs.

APNs is a critical part of configuring and securing mobile devices. If there are things that prevent the direct and persistent connection to and from APNs then the entire MDM management capabilities will be inoperative.

APNs Networking Considerations

When Clients and MDM servers are behind a firewall, some network configuration may need to take place in order for the MDM server and device communication to function properly.

- Outbound traffic rules will allow the MDM server and clients to establish communication outbound from the host network, but will not allow traffic outside the host network to establish inbound communication to the MDM server, or clients.
- All communication will be established outbound, and should be allowed to bypass content filters, proxies, and firewalls to communicate directly with APNs.
- Though the communication is established outbound from the MDM server and from client devices, once connected to the APNs service, bi-directional communication will flow via that connection between the MDM server, clients and Apple.
- APNs uses a load balancing scheme that prevents static whitelisting of IP/port. As the entire 17.0.0.0/8 address block is assigned to Apple, firewall rules can be established to specify that range.

Clients and MDM servers require a direct connection to Apple's Push Notification Service (APNs).

A proxy server on the Wi-Fi network will prevent devices from being able to use APNs, because APNs requires a direct and persistent connection. Therefore, you will need to make the indicated ports open on your proxy server to guarantee the connection to APNs.

Some educational institutions that use ISP Co-Ops may have additional filters and should be provided this documentation.

Server (JSS)

APNs Server

Client Devices

Clients

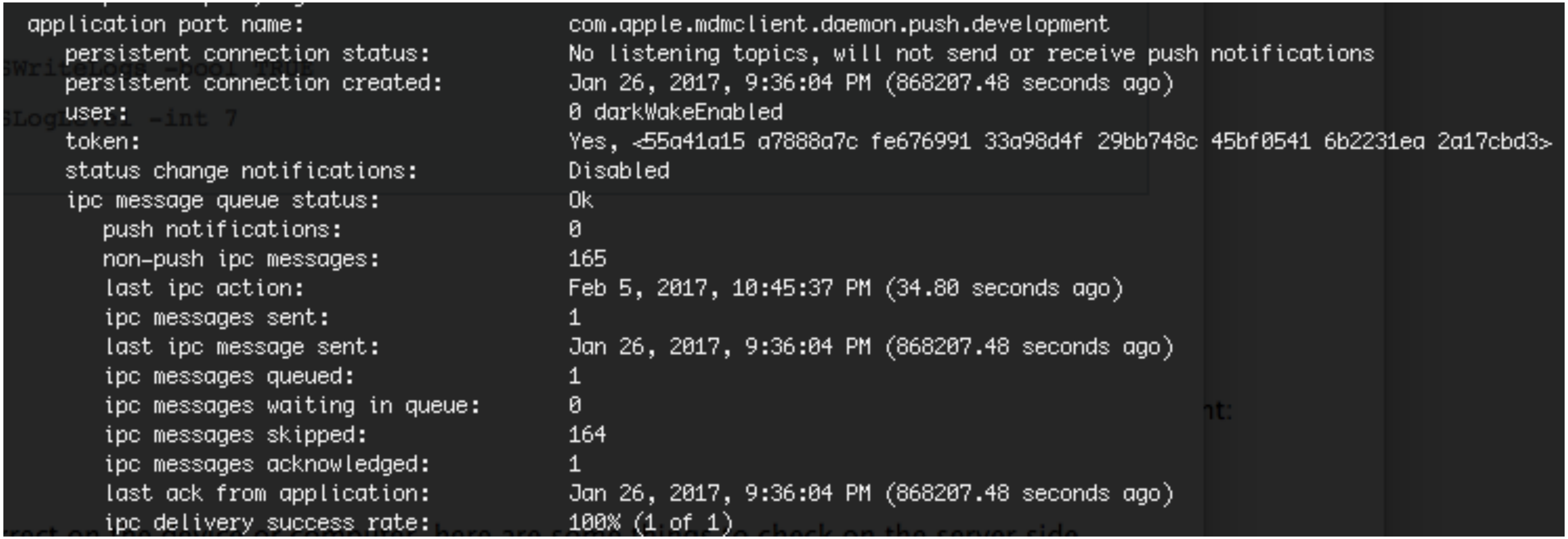
To ensure reliable client side communication, allow outbound connections to Apple's 17.0.0.0/8 block, over TCP port 5223, and 443.

MDM Server

To ensure reliable server communication, allow outbound connections to Apple's 17.0.0.0/8 block over TCP port 2195 and 2196.

https://www.jamf.com/resources/making-apple-push-notification-service-available-on-your-network/

/System/Library/PrivateFrameworks/ApplePushService.framework/apsctl status



```
application port name:                   com.apple.mdmclient.daemon.push.development
   persistent connection status:         No listening topics, will not send or receive push notifications
   persistent connection created:        Jan 26, 2017, 9:36:04 PM (868207.48 seconds ago)
   user:                                 0 darkWakeEnabled
   token:                                Yes, <55a41a15 a7888a7c fe676991 33a98d4f 29bb748c 45bf0541 6b2231ea 2a17cbd3>
   status change notifications:          Disabled
   ipc message queue status:             Ok
      push notifications:                0
      non-push ipc messages:             165
      last ipc action:                   Feb 5, 2017, 10:45:37 PM (34.80 seconds ago)
      ipc messages sent:                 1
      last ipc message sent:             Jan 26, 2017, 9:36:04 PM (868207.48 seconds ago)
      ipc messages queued:               1
      ipc messages waiting in queue:     0
      ipc messages skipped:              164
      ipc messages acknowledged:         1
      last ack from application:         Jan 26, 2017, 9:36:04 PM (868207.48 seconds ago)
      ipc delivery success rate:         100% (1 of 1)
```

Why can’t I use my proxy server?

Certificate Pinning

https://www.bluecoat.com/ko/documents/download/7ff09c94-7b88-4319-a766-191c9dedde22

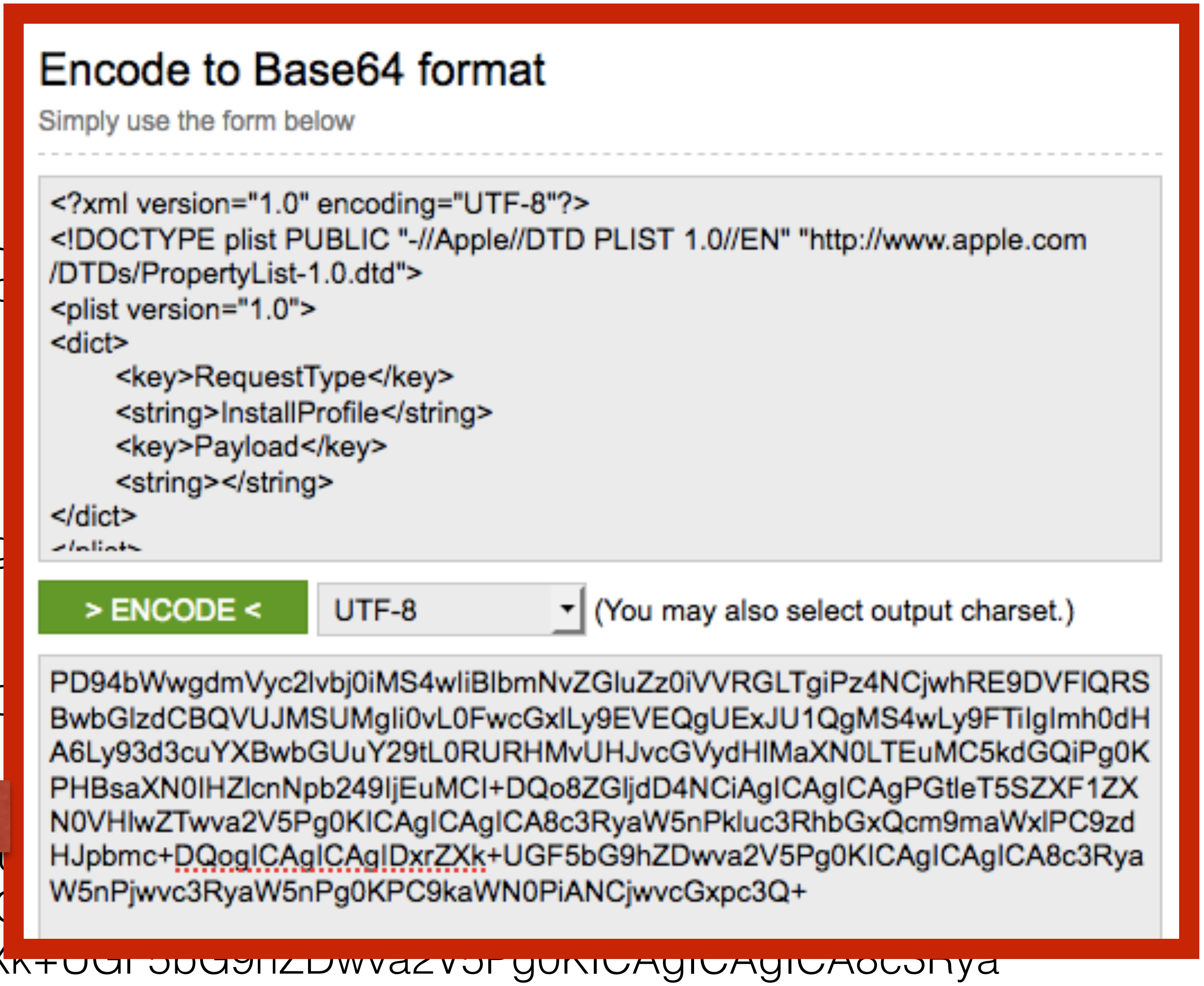
Is that the same for all vendors?

Yes

If I don’t open ports to the MDM Server?

Webhook on MDM Server

```
<?xml version="1.0" encoding="U
<!DOCTYPE plist PUBLIC "-//App
PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
        <key>RequestType</key>
        <string>InstallProfile</string
        <key>Payload</key>

<string>PD94bWwgdmVyc2lvbj0
FIQRSBwbGlzdCBQV      MSUMg
h0dHA6Ly93d3cuY
g0KPHBsaXN0IHZlc     249IjE
ZXN0VHlwZTwva2V5Pg  KICAgI
HJpbmc+DQogICAgICAgIDxrZXk+UGF5bG9hZDwva2V5Pg0KICAgICAgICA8c3Rya
W5nPjwvc3RyaW5nPg0KPC9kaWN0PiANCjwvcGxpc3Q+</string>
</dict>
</plist>
```



Encode to Base64 format

Simply use the form below

```
<?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>RequestType</key>
        <string>InstallProfile</string>
        <key>Payload</key>
        <string></string>
</dict>
</plist>
```

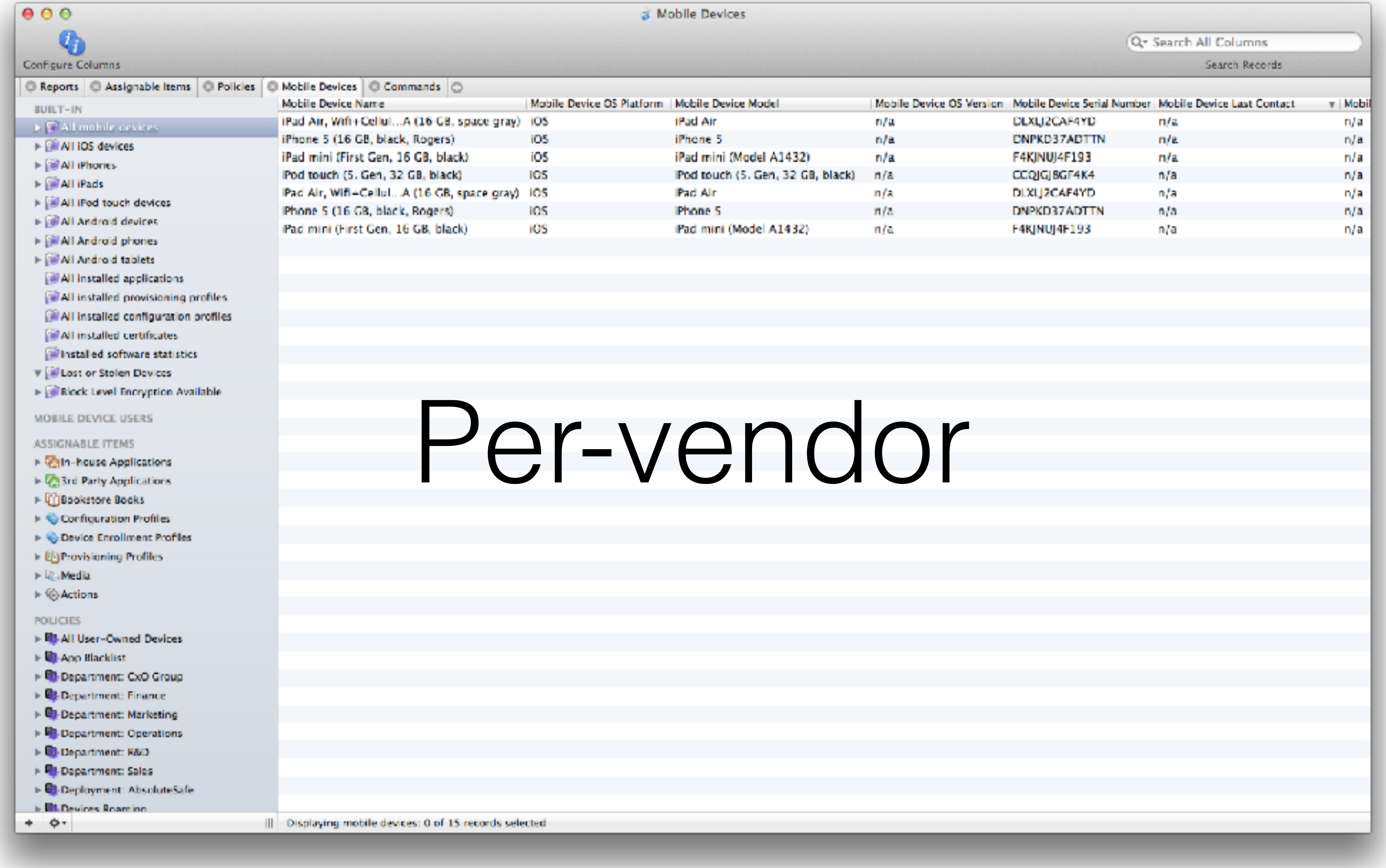
(You may also select output charset.)

```
PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4NCjwhRE9DVFlQRS
BwbGlzdCBQVUJMSUMgIi0vL0FwcGxlLy9EVEQgUExJU1QgMS4wLy9FTiIgImh0dH
A6Ly93d3cuYXBwbGUuY29tL0RURHMvUHJvcGVydHlMaXN0LTEuMC5kdGQiPg0K
PHBsaXN0IHZlcnNpb249IjEuMCI+DQo8ZGljdD4NCiAgICAgICAgPGtleT5SZXF1ZX
N0VHlwZTwva2V5Pg0KICAgICAgICA8c3RyaW5nPkluc3RhbGxQcm9maWxlPC9zd
HJpbmc+DQogICAgICAgIDxrZXk+UGF5bG9hZDwva2V5Pg0KICAgICAgICA8c3Rya
W5nPjwvc3RyaW5nPg0KPC9kaWN0PiANCjwvcGxpc3Q+
```

Do I need SCEP?

SCEP

- Device uses SCEP to obtain a cert and then communicates that cert back to us during enrollment
- Each client receives a unique cert
- If certs are from SCEP they should be unique
    - *Can install SCEP payloads with a profile*



Mobile Device Name	Mobile Device OS Platform	Mobile Device Model	Mobile Device OS Version	Mobile Device Serial Number	Mobile Device Last Contact	Mobil
iPad Air, Wifi+Cellul...A (16 GB, space gray)	iOS	iPad Air	n/a	DLXLJ2CAF4YD	n/a	n/a
iPhone 5 (16 GB, black, Rogers)	iOS	iPhone 5	n/a	DNPKD37ADTTN	n/a	n/a
iPad mini (First Gen, 16 GB, black)	iOS	iPad mini (Model A1432)	n/a	F4KJNUJ4F193	n/a	n/a
iPod touch (5. Gen, 32 GB, black)	iOS	iPod touch (5. Gen, 32 GB, black)	n/a	CCQJGJBGF4K4	n/a	n/a
iPad Air, Wifi+Cellul...A (16 GB, space gray)	iOS	iPad Air	n/a	DLXLJ2CAF4YD	n/a	n/a
iPhone 5 (16 GB, black, Rogers)	iOS	iPhone 5	n/a	DNPKD37ADTTN	n/a	n/a
iPad mini (First Gen, 16 GB, black)	iOS	iPad mini (Model A1432)	n/a	F4KJNUJ4F193	n/a	n/a

Displaying mobile devices: 0 of 15 records selected

Per-vendor

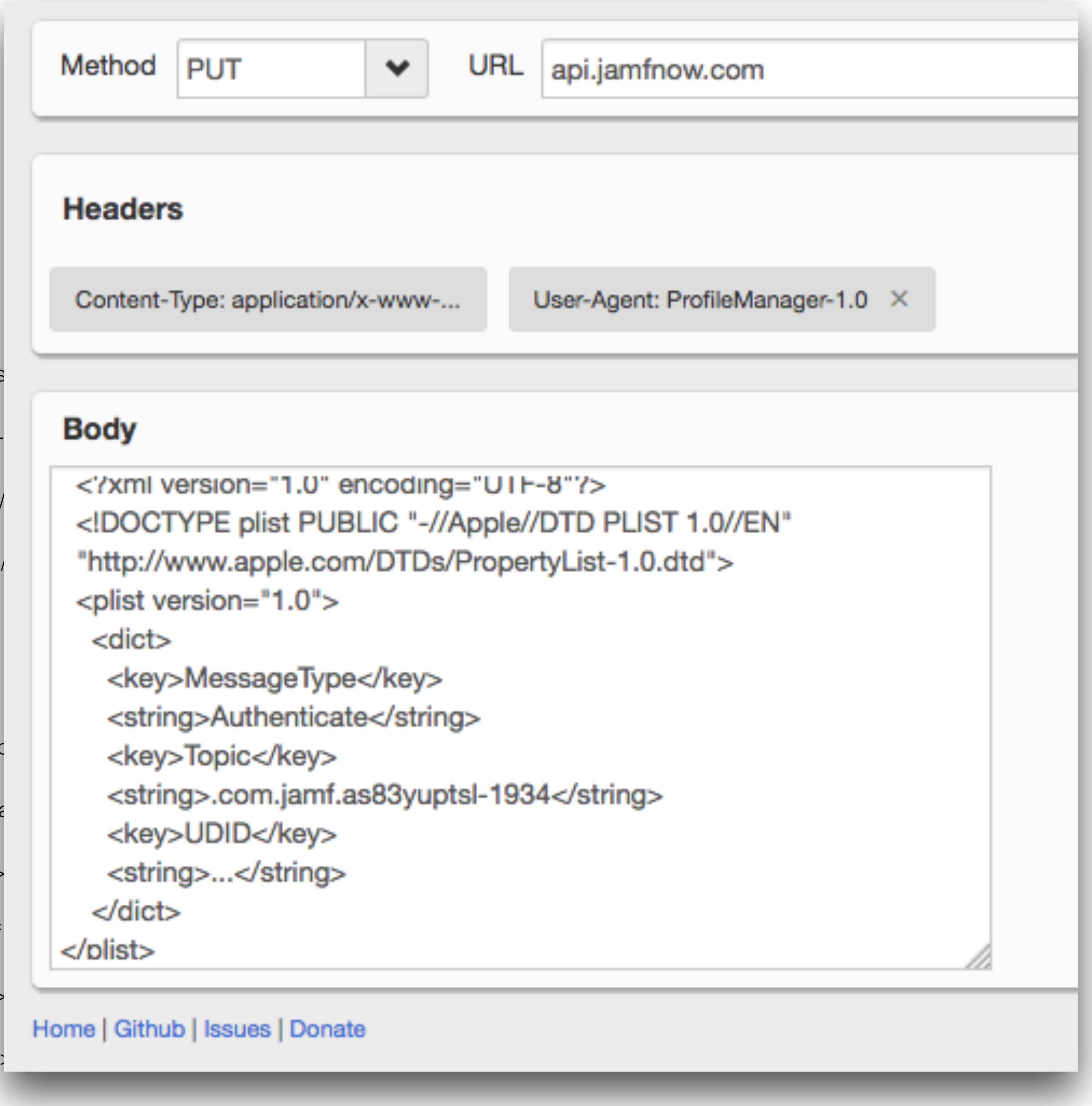
What if devices fail to enroll?

The MDM Check-In Command

https://developer.apple.com/library/content/documentation/Miscellaneous/Reference/MobileDeviceManagementProtocolRef/2-MDM_Check_In_Protocol/MDM_Check_In_Protocol..html

What a Check-in Request

```
PUT api.jamfnow.com HTTP/1.1

  Host: jamfnow.com

  Content-Length: 1234

  Content-Type: application/x-apple-as

  <?xml version="1.0" encoding="UTF-

        <!DOCTYPE plist PUBLIC "-//

        "http://www.apple.com/DTDs/

        <plist version="1.0">

         <dict>

                 <key>MessageTyp

                 <string>Authentica

                 <key>Topic</key>

                 <string>.com.jamf

                 <key>UDID</key>

                 <string>...</string

         </dict>

</plist>
```



Method PUT

URL api.jamfnow.com

Headers

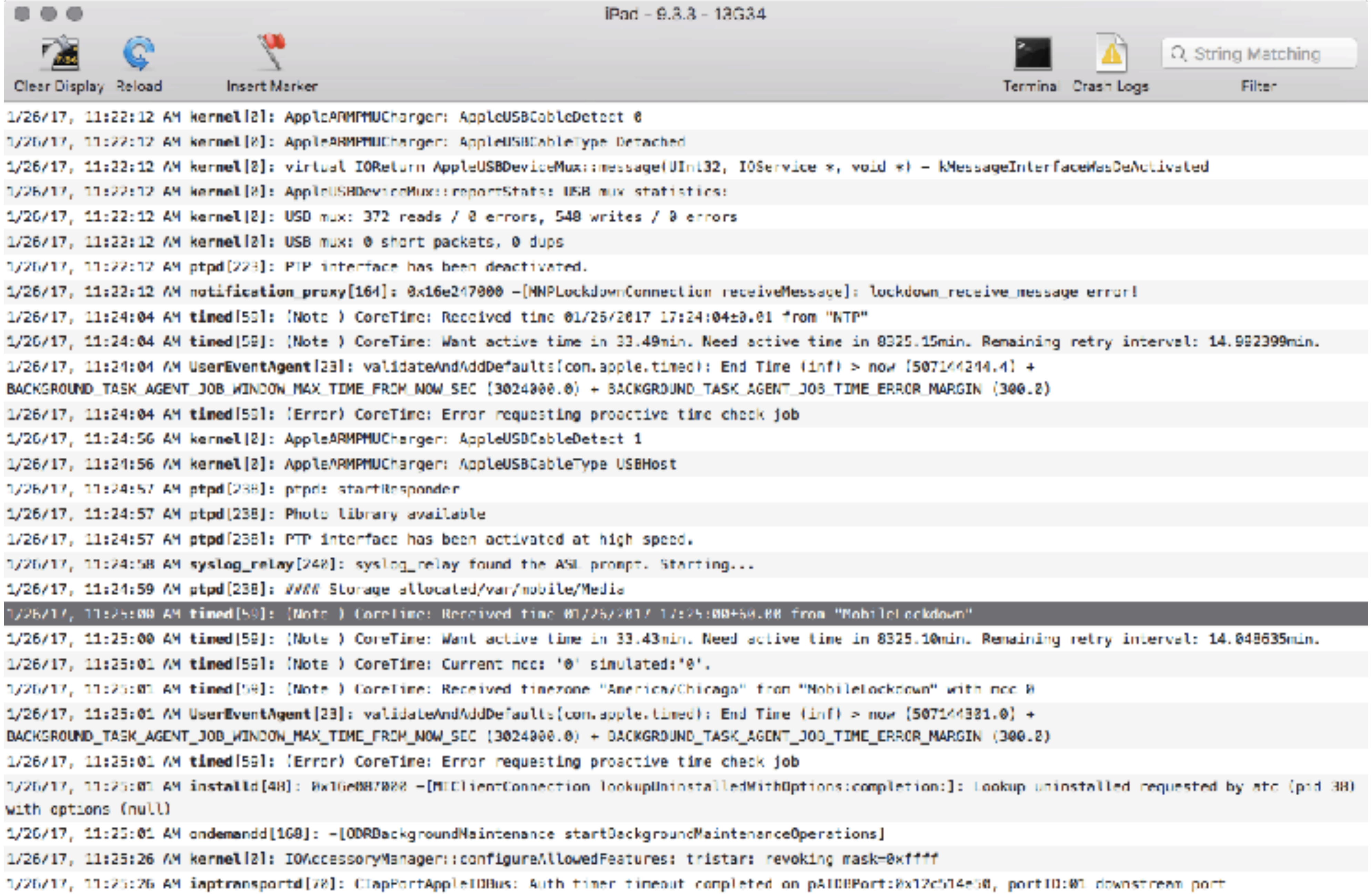
Content-Type: application/x-www-...

User-Agent: ProfileManager-1.0 ×

Body

```
<?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>MessageType</key>
    <string>Authenticate</string>
    <key>Topic</key>
    <string>.com.jamf.as83yuptsl-1934</string>
    <key>UDID</key>
    <string>...</string>
  </dict>
</plist>
```

Use iOS Console To View Transactions



```
1/26/17, 11:22:12 AM kernel[0]: AppleARMPMUCharger: AppleUSBCableDetect 0
1/26/17, 11:22:12 AM kernel[0]: AppleARMPMUCharger: AppleUSBCableType Detached
1/26/17, 11:22:12 AM kernel[0]: virtual IOReturn AppleUSBDeviceMux::message(UInt32, IOService *, void *) - kMessageInterfaceWasDeActivated
1/26/17, 11:22:12 AM kernel[0]: AppleUSBDeviceMux::reportStats: USB mux statistics:
1/26/17, 11:22:12 AM kernel[0]: USB mux: 372 reads / 0 errors, 548 writes / 0 errors
1/26/17, 11:22:12 AM kernel[0]: USB mux: 0 short packets, 0 dups
1/26/17, 11:22:12 AM ptpd[223]: PTP interface has been deactivated.
1/26/17, 11:22:12 AM notification_proxy[164]: 0x16e247000 -[MNPLockdownConnection receiveMessage]: lockdown_receive_message error!
1/26/17, 11:24:04 AM timed[59]: (Note ) CoreTime: Received time 01/26/2017 17:24:04±0.01 from "NTP"
1/26/17, 11:24:04 AM timed[59]: (Note ) CoreTime: Want active time in 33.49min. Need active time in 8325.15min. Remaining retry interval: 14.992399min.
1/26/17, 11:24:04 AM UserEventAgent[23]: validateAndAddDefaults(com.apple.timed): End Time (inf) > now (507144244.4) +
BACKGROUND_TASK_AGENT_JOB_WINDOW_MAX_TIME_FROM_NOW_SEC (3024000.0) + BACKGROUND_TASK_AGENT_JOB_TIME_ERROR_MARGIN (300.2)
1/26/17, 11:24:04 AM timed[59]: (Error) CoreTime: Error requesting proactive time check job
1/26/17, 11:24:56 AM kernel[0]: AppleARMPMUCharger: AppleUSBCableDetect 1
1/26/17, 11:24:56 AM kernel[0]: AppleARMPMUCharger: AppleUSBCableType USBHost
1/26/17, 11:24:57 AM ptpd[238]: ptpd: startResponder
1/26/17, 11:24:57 AM ptpd[238]: Photo library available
1/26/17, 11:24:57 AM ptpd[238]: PTP interface has been activated at high speed.
1/26/17, 11:24:58 AM syslog_relay[248]: syslog_relay found the ASL prompt. Starting...
1/26/17, 11:24:59 AM ptpd[238]: WWW Storage allocated/var/mobile/Media
1/26/17, 11:25:00 AM timed[59]: (Note ) CoreTime: Received time 01/26/2017 17:25:00±50.00 from "MobileLockdown"
1/26/17, 11:25:00 AM timed[59]: (Note ) CoreTime: Want active time in 33.43min. Need active time in 8325.10min. Remaining retry interval: 14.048635min.
1/26/17, 11:25:01 AM timed[59]: (Note ) CoreTime: Current mcc: '0' simulated:'0'.
1/26/17, 11:25:01 AM timed[59]: (Note ) CoreTime: Received timezone "America/Chicago" from "MobileLockdown" with mcc 0
1/26/17, 11:25:01 AM UserEventAgent[23]: validateAndAddDefaults(com.apple.timed): End Time (inf) > now (507144301.0) +
BACKGROUND_TASK_AGENT_JOB_WINDOW_MAX_TIME_FROM_NOW_SEC (3024000.0) + BACKGROUND_TASK_AGENT_JOB_TIME_ERROR_MARGIN (300.2)
1/26/17, 11:25:01 AM timed[59]: (Error) CoreTime: Error requesting proactive time check job
1/26/17, 11:25:01 AM installd[48]: 0x16e087000 -[MIClientConnection lookupUninstalledWithOptions:completion:]: Lookup uninstalled requested by atc (pid 38)
with options (null)
1/26/17, 11:25:01 AM ondemandd[168]: -[ODRBackgroundMaintenance startBackgroundMaintenanceOperations]
1/26/17, 11:25:26 AM kernel[0]: IOAccessoryManager::configureAllowedFeatures: tristar: revoking mask=0xffff
1/26/17, 11:25:26 AM iaptransportd[78]: CIapPortAppleIDBus: Auth timer timeout completed on pAIDBPort:0x12c514e50, portID:01 downstream port
```

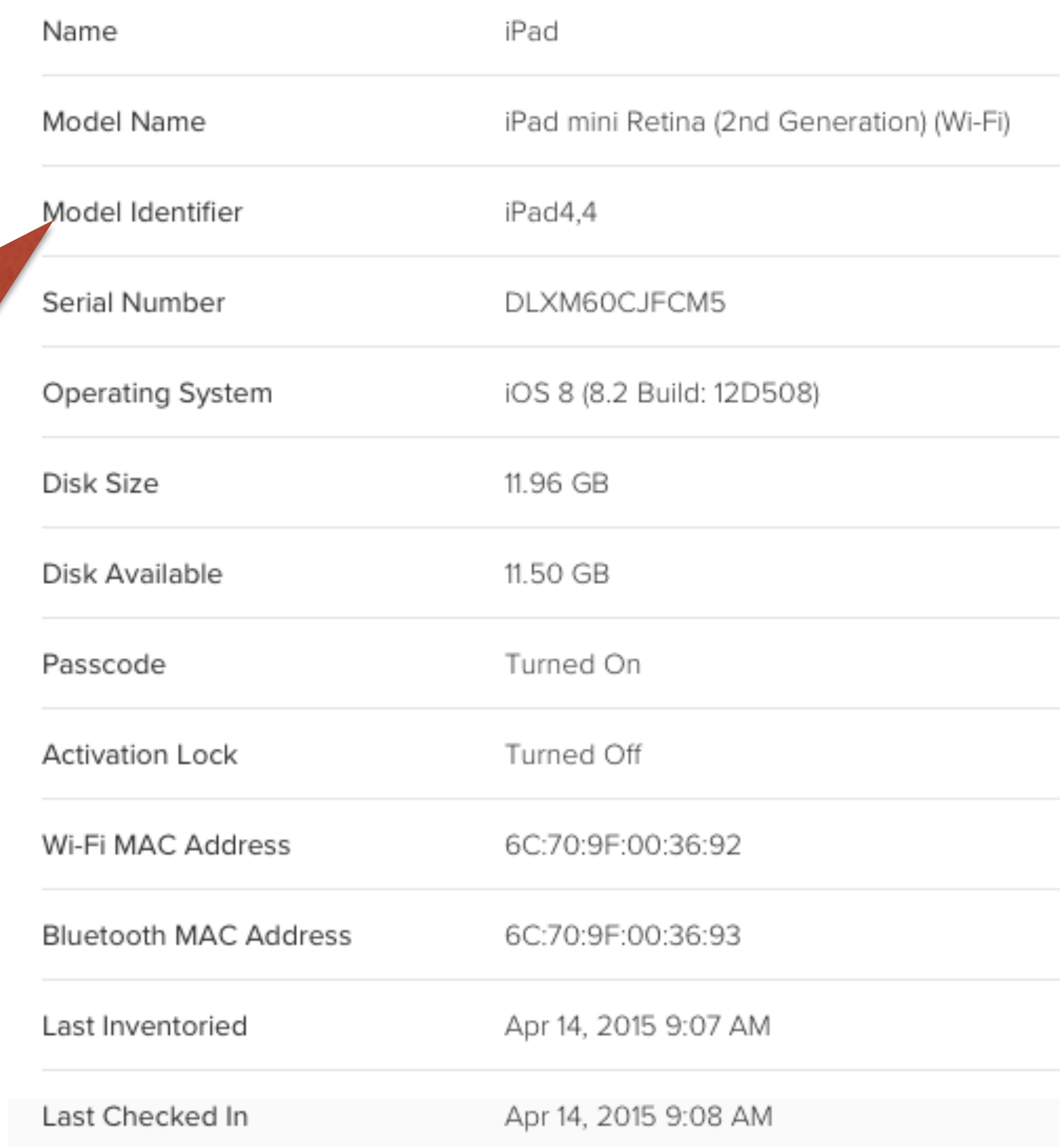
https://lemonjar.com/iosconsole/

Authenticate

Verify that a device can enroll

Authenticate

- MessageType: Authenticate
- Topic
- UDID
- OSVersion
- BuildVersion
- ProductName
- SerialNumber
- IMEI
- MEID



Name	iPad
Model Name	iPad mini Retina (2nd Generation) (Wi-Fi)
Model Identifier	iPad4,4
Serial Number	DLXM60CJFCM5
Operating System	iOS 8 (8.2 Build: 12D508)
Disk Size	11.96 GB
Disk Available	11.50 GB
Passcode	Turned On
Activation Lock	Turned Off
Wi-Fi MAC Address	6C:70:9F:00:36:92
Bluetooth MAC Address	6C:70:9F:00:36:93
Last Inventoried	Apr 14, 2015 9:07 AM
Last Checked In	Apr 14, 2015 9:08 AM

200 = Success
401 = Failure

What if a device stops responding to MDM commands?

TokenUpdate

Updates token used to communicate with server (push magic and APNs token)

TokenUpdate

- MessageType: TokenUpdate
- Topic (must match push notification cert)
- UDID
- Token
- PushMagic
- UnlockToken
- Awaiting-Configuration (for DEP - send commands during bootstrap)

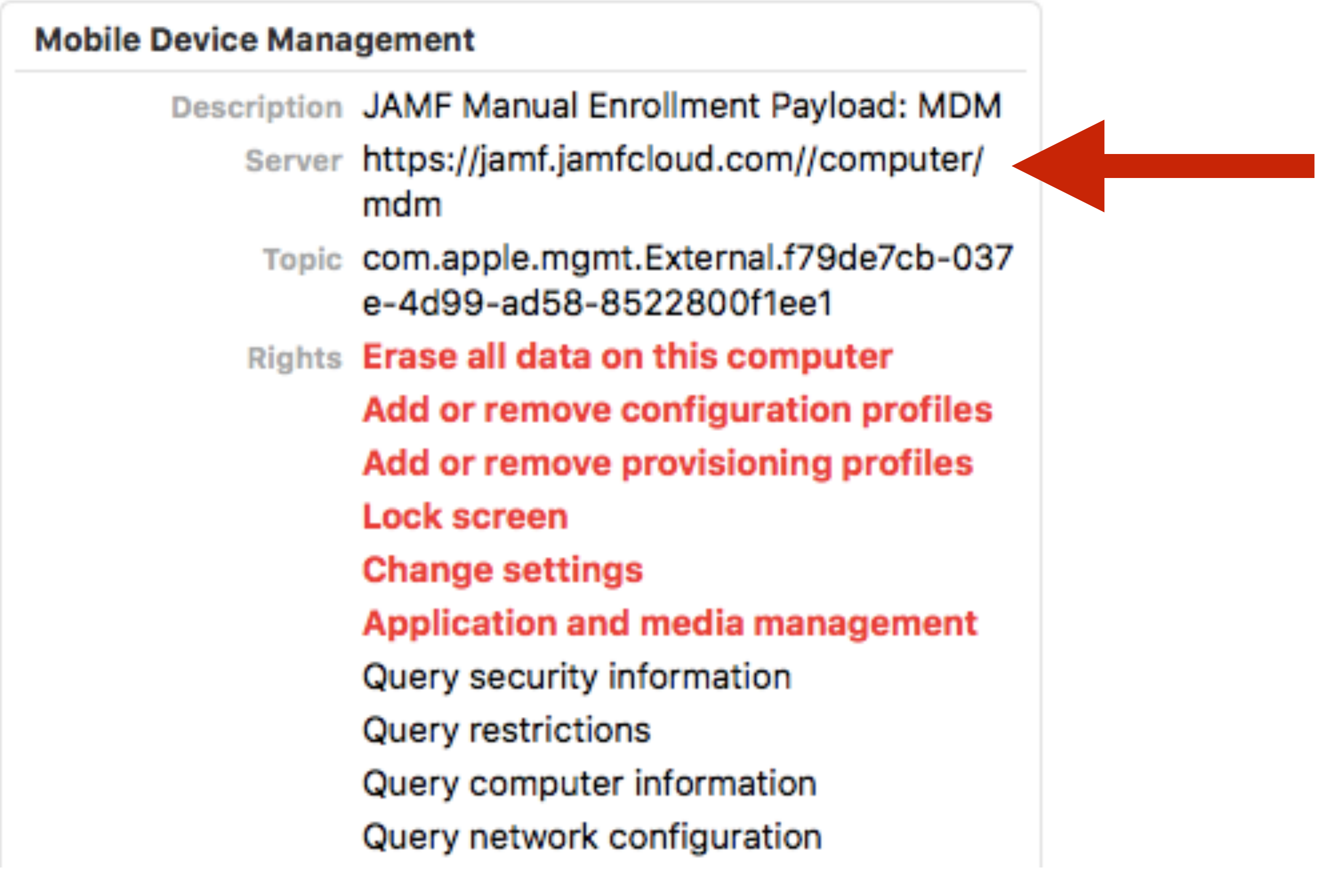
CheckOut

Device sends a command that it’s leaving management

CheckOut

- Best effort…
  - MessageType: CheckOut
  - Topic
  - UDID

Can I change the URL of my MDM Server?



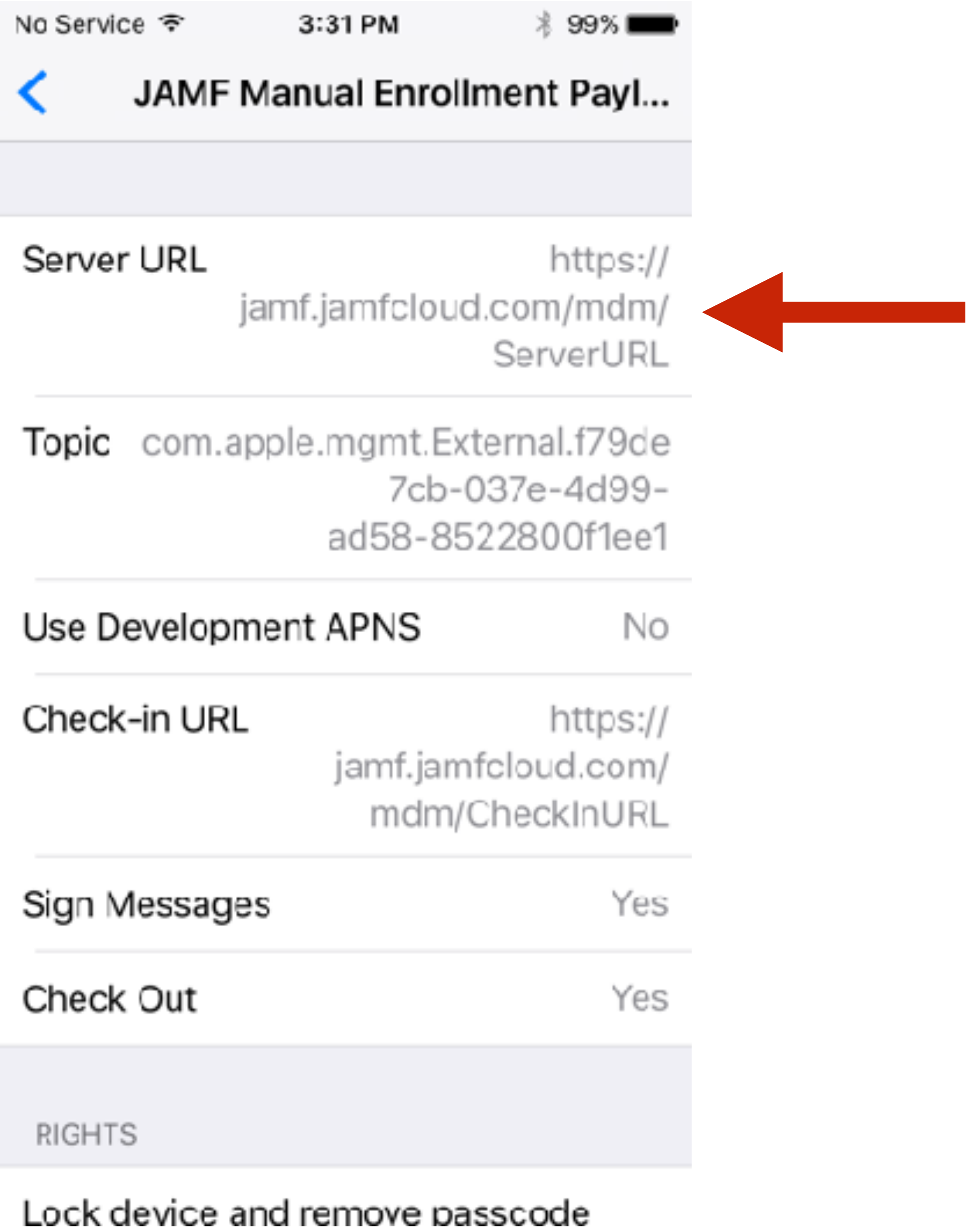
Mobile Device Management

Description JAMF Manual Enrollment Payload: MDM

Server https://jamf.jamfcloud.com//computer/mdm

Topic com.apple.mgmt.External.f79de7cb-037e-4d99-ad58-8522800f1ee1

Rights **Erase all data on this computer**
Add or remove configuration profiles
Add or remove provisioning profiles
Lock screen
Change settings
Application and media management
Query security information
Query restrictions
Query computer information
Query network configuration



JAMF Manual Enrollment Payl...

Server URL https://jamf.jamfcloud.com/mdm/ServerURL

Topic com.apple.mgmt.External.f79de7cb-037e-4d99-ad58-8522800f1ee1

Use Development APNS No

Check-in URL https://jamf.jamfcloud.com/mdm/CheckInURL

Sign Messages Yes

Check Out Yes

RIGHTS

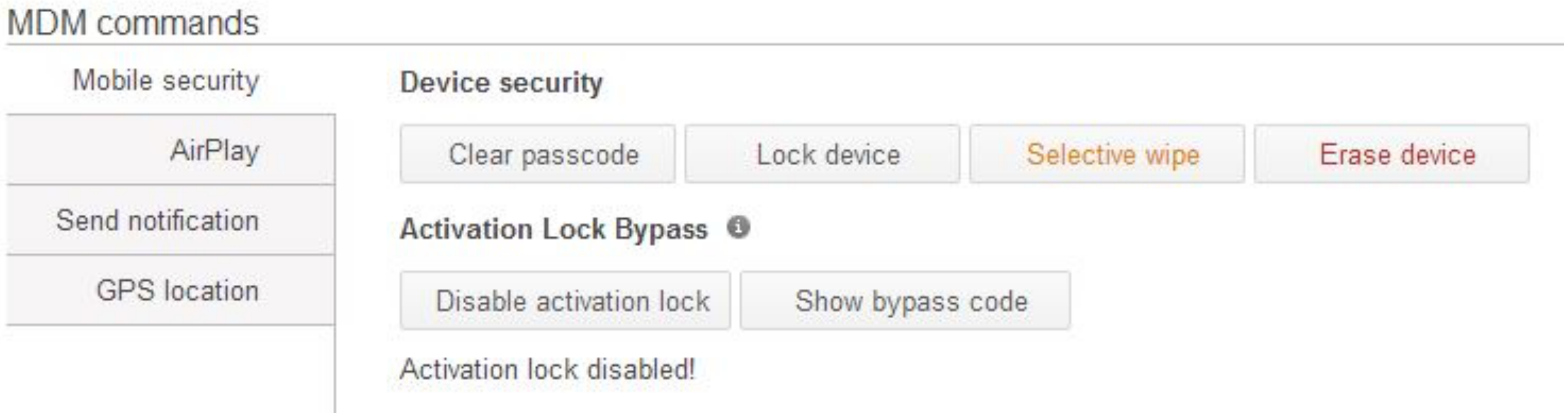
Lock device and remove passcode

Commands

Activation Lock Bypass

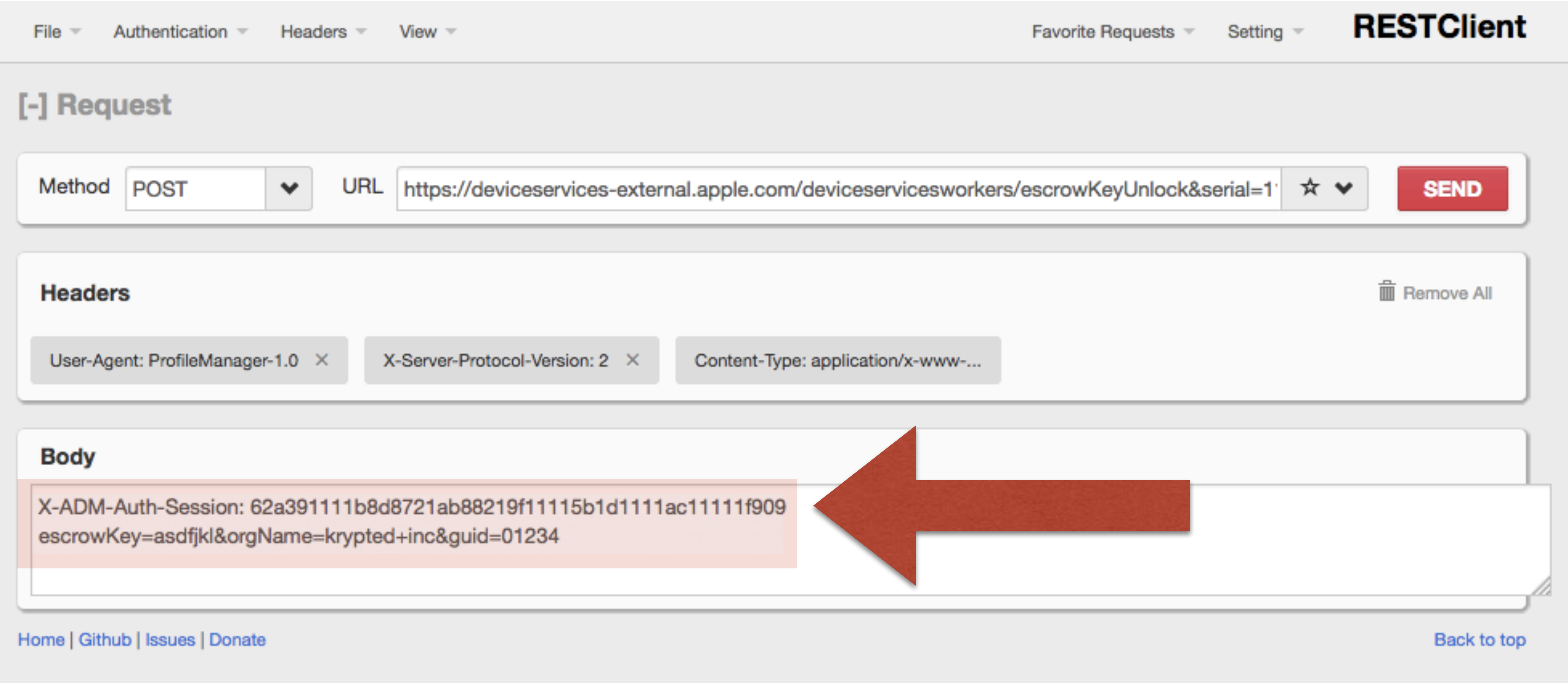
EscrowKeyUnlock

FileWave



Activation lock disabled!

X-ADM-Auth-Session



Method: POST

URL: https://deviceservices-external.apple.com/deviceservicesworkers/escrowKeyUnlock&serial=1

Headers

User-Agent: ProfileManager-1.0 ×

X-Server-Protocol-Version: 2 ×

Content-Type: application/x-www-...

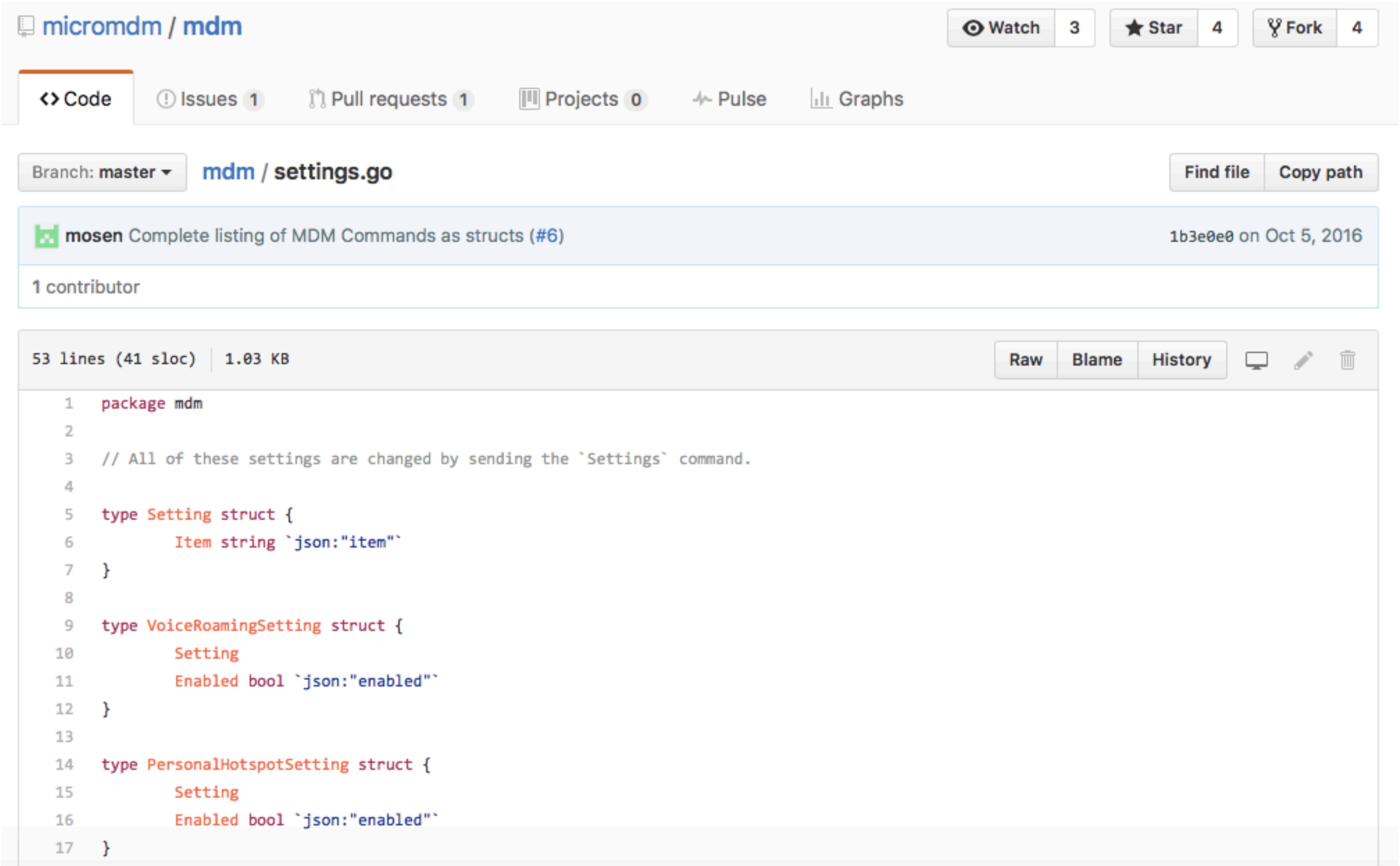
Body

```
X-ADM-Auth-Session: 62a391111b8d8721ab88219f11115b1d1111ac11111f909
escrowKey=asdfjkl&orgName=krypted+inc&guid=01234
```

How’d we get that code?

- ActivationLockBypassCode
- Obtained at enrollment
- If Supervised
- Then you can EscrowKeyUnlock

Settings



mdm / settings.go

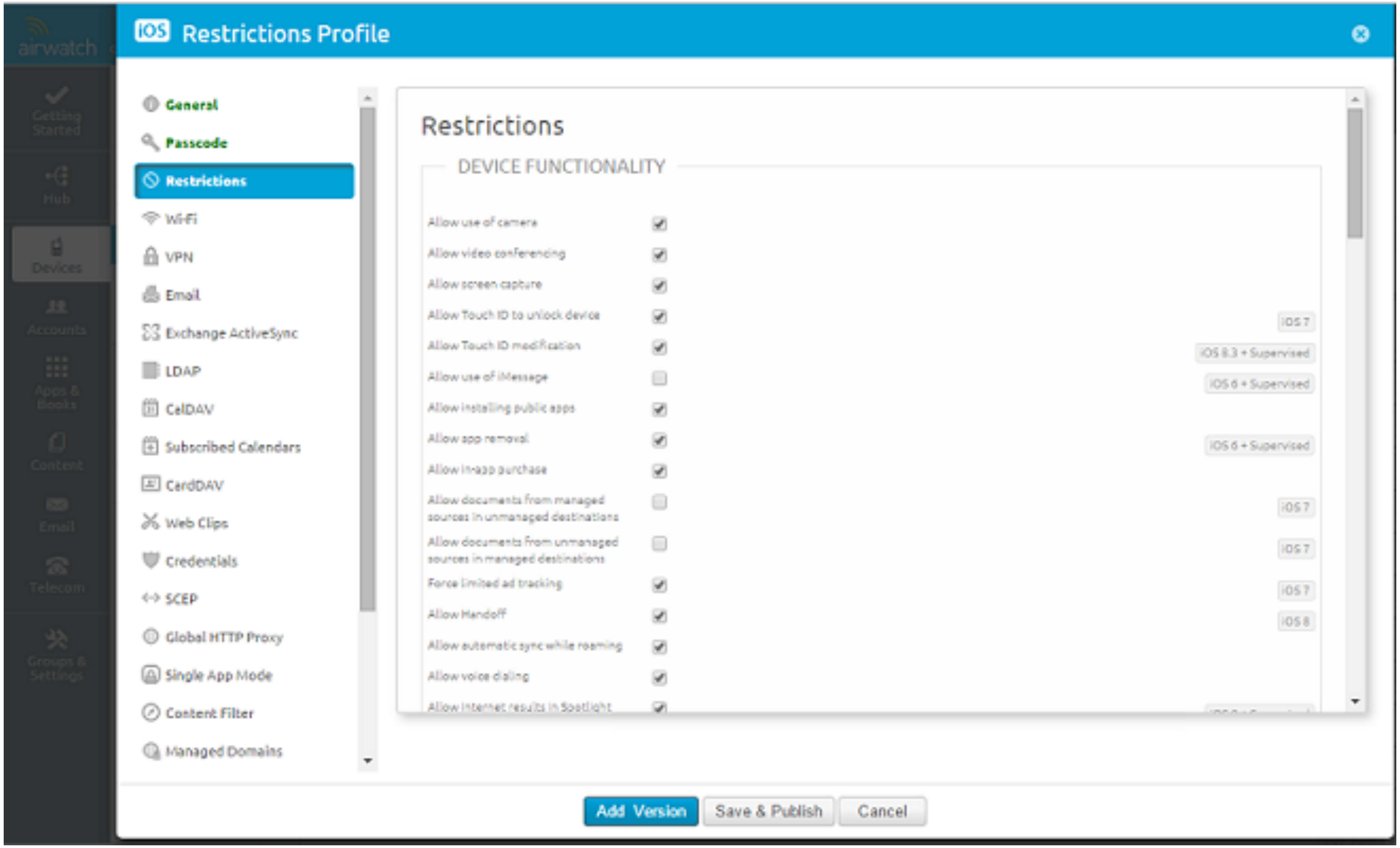
mosen Complete listing of MDM Commands as structs (#6) 1b3e0e0 on Oct 5, 2016

1 contributor

53 lines (41 sloc) 1.03 KB

```go
package mdm

// All of these settings are changed by sending the `Settings` command.

type Setting struct {
	Item string `json:"item"`
}

type VoiceRoamingSetting struct {
	Setting
	Enabled bool `json:"enabled"`
}

type PersonalHotspotSetting struct {
	Setting
	Enabled bool `json:"enabled"`
}
```

AirWatch Profiles



Restrictions

DEVICE FUNCTIONALITY

Allow use of camera ☑

Allow video conferencing ☑

Allow screen capture ☑

Allow Touch ID to unlock device ☑ iOS 7

Allow Touch ID modification ☑ iOS 8.3 + Supervised

Allow use of iMessage □ iOS 6 + Supervised

Allow installing public apps ☑

Allow app removal ☑ iOS 6 + Supervised

Allow in-app purchase ☑

Allow documents from managed sources in unmanaged destinations □ iOS 7

Allow documents from unmanaged sources in managed destinations □ iOS 7

Force limited ad tracking ☑ iOS 7

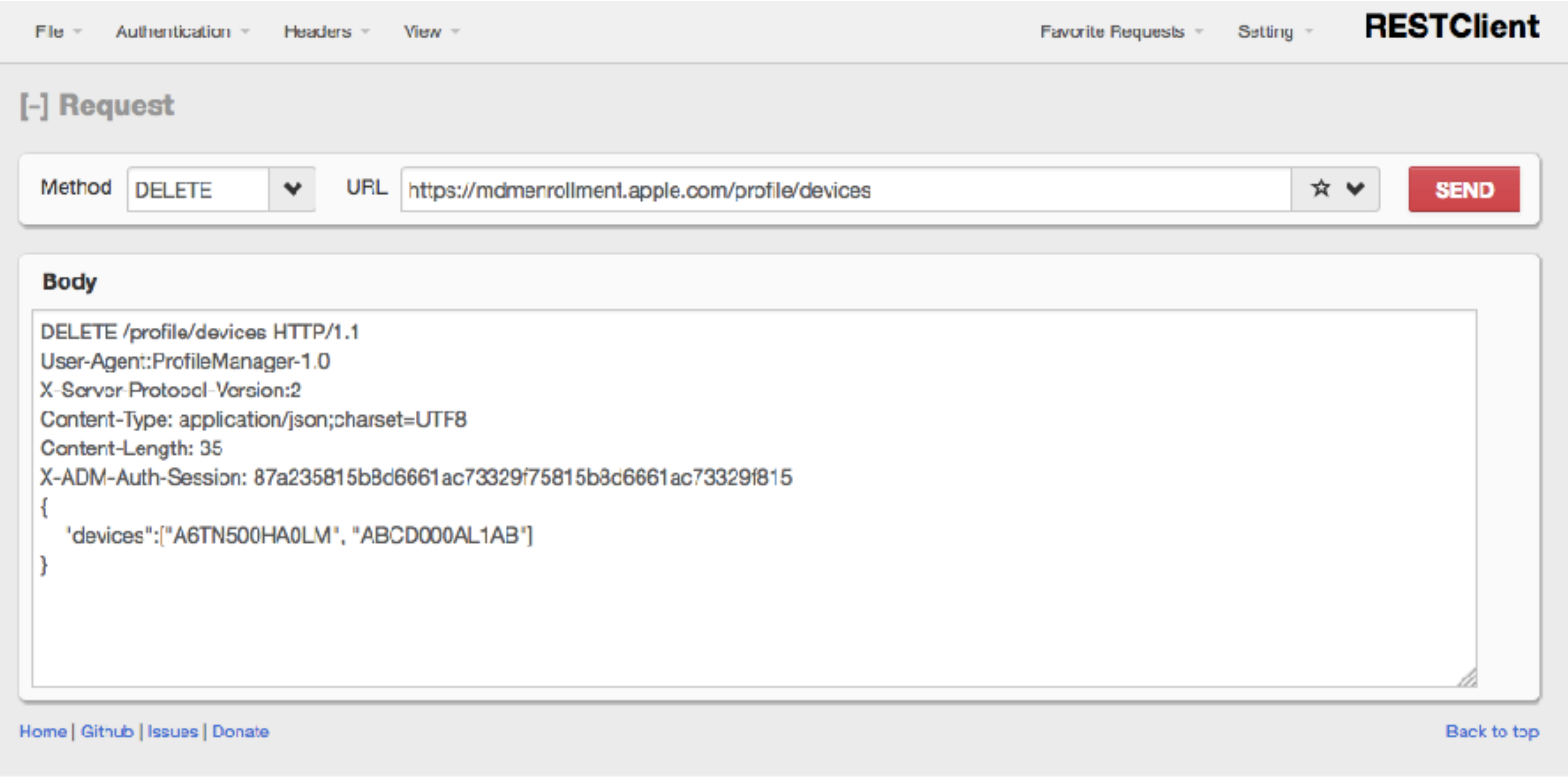
Allow Handoff ☑ iOS 8

Allow automatic sync while roaming ☑

Allow voice dialing ☑

Allow internet results in Spotlight ☑

Delete Profiles



Method DELETE

URL https://mdmenrollment.apple.com/profile/devices

```
DELETE /profile/devices HTTP/1.1
User-Agent::ProfileManager-1.0
X-Server-Protocol-Version:2
Content-Type: application/json;charset=UTF8
Content-Length: 35
X-ADM-Auth-Session: 87a235815b8d6661ac73329f75815b8d6661ac73329f815
{
   'devices":["A6TN500HA0LM", "ABCD000AL1AB"]
}
```

Does the MDM inventory contain app information?

VPP

What’s In The stoken

Stoken

- eyJ0b2tlbuKAnTrigJ1hYWFhUnpwTEV0YWFhYStuc3hDZH
dyY3QwUmp3ZGljTmFhYWFUWXE4VVAyc2hSYTBMUnVG
cVpQM0pLQmJUTWxDSE42ZzNtc1J6WVlQbVVkVXJBS2x
3PT0iLCJleHBEYXRlIjoiMjAxNi0wNC0yMVQxMjowNzozMi0
wNzAwIiwib3JnTmFtZeKAnTrigJ1rcnlwdGVkLjIwMTAxMTE
4MDAifQ==

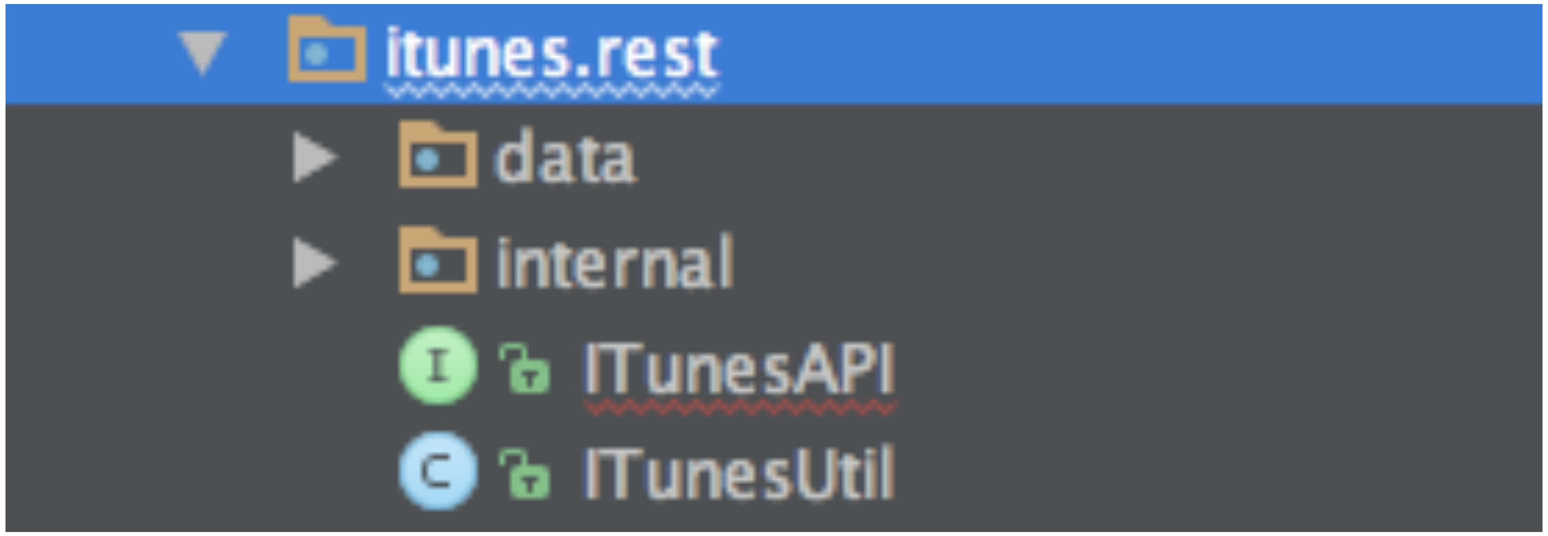
- base64 -i stoken

- {"token”:”aaaaRzpLEtaaaa+nsxCdwrct0RjwdicNaaaaTYq8
UP2shRa0LRuFqZP3JKBbTMlCHN6g3msRzYYPmUdUrAK
lw==","expDate":"2016-04-21T12:07:32-0700","orgName”:”
krypted.2010111800"}

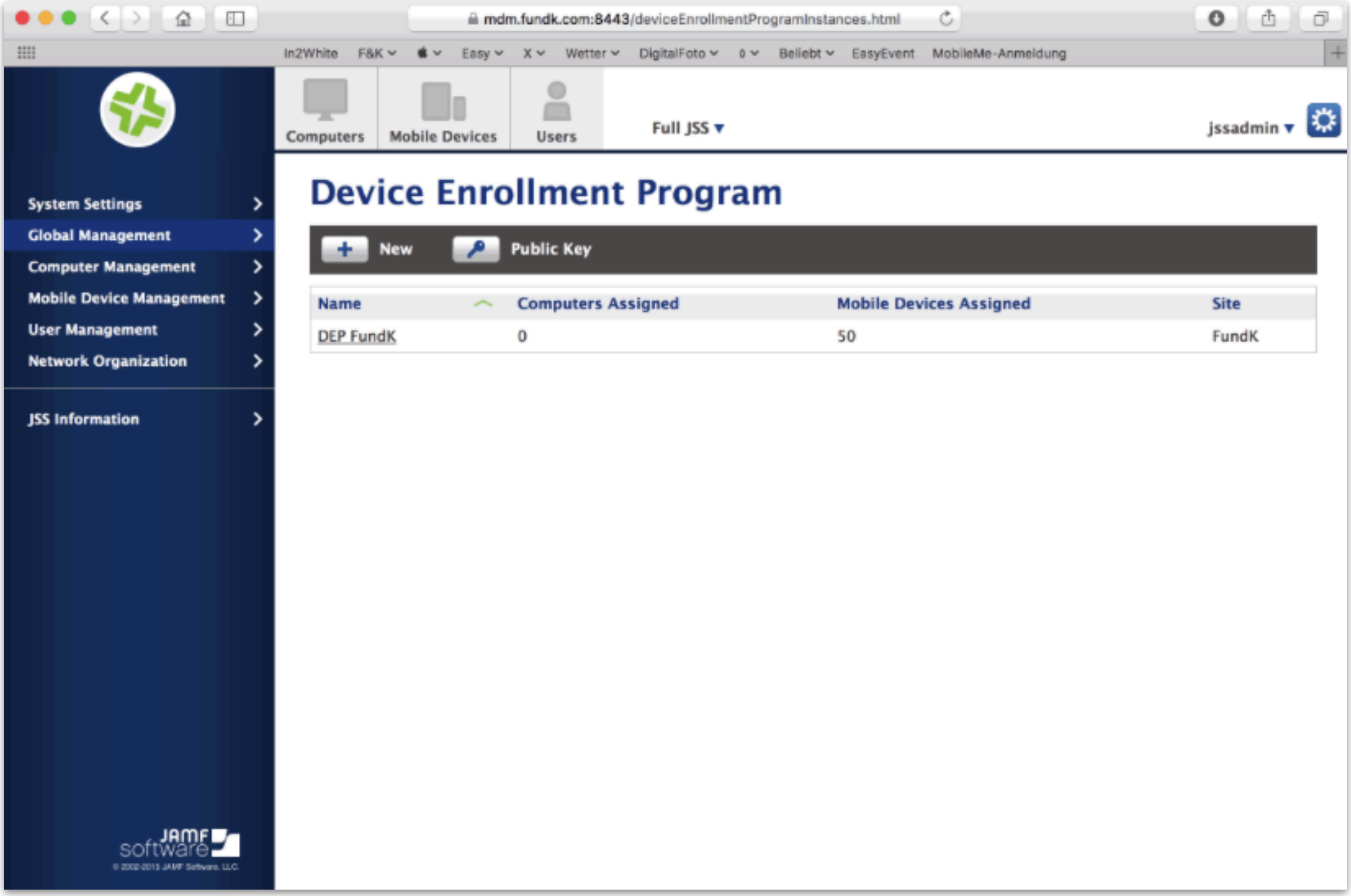
The VPP Service

- Mostly per-device and per-user
- Some places buy 10k copies of free apps
- Syncs all data back
- For privacy, VPP endpoint doesn’t know which user is which (we get a hash)
- If the service isn’t available a GUI might go unresponsive

Polling VPP Is Weird



Who wants to talk about DEP?



Device Enrollment Program

Name	Computers Assigned	Mobile Devices Assigned	Site
DEP FundK	0	50	FundK

Best Practices

Make sure to open those ports

Use Profile Manager
For Comparison Testing

No profile conflicts

Who’s enrolling?!?!

Use libimobiledevice

http://krypted.com/uncategorized/command-line-ios-device-management/

Resources

- MDM Protocol Reference: https://developer.apple.com/library/content/documentation/Miscellaneous/Reference/MobileDeviceManagementProtocolRef/1-Introduction/Introduction.html#//apple_ref/doc/uid/TP40017387-CH1-SW1

- Security Concepts: https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/iPhoneOTAConfiguration/OTASecurity/OTASecurity.html

- MicroMDM: https://github.com/micromdm

Resources

- Enhanced APNs API: https://developer.apple.com/library/ios/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/Chapters/APNsProviderAPI.html#//apple_ref/doc/uid/TP40008194-CH101-SW1

- enterpriseios.com

Client-side configuration options

```
defaults write /Library/Preferences/com.apple.mdmclient
BypassPreLoginCheck -bool YES
```

Q&A

Thank you!