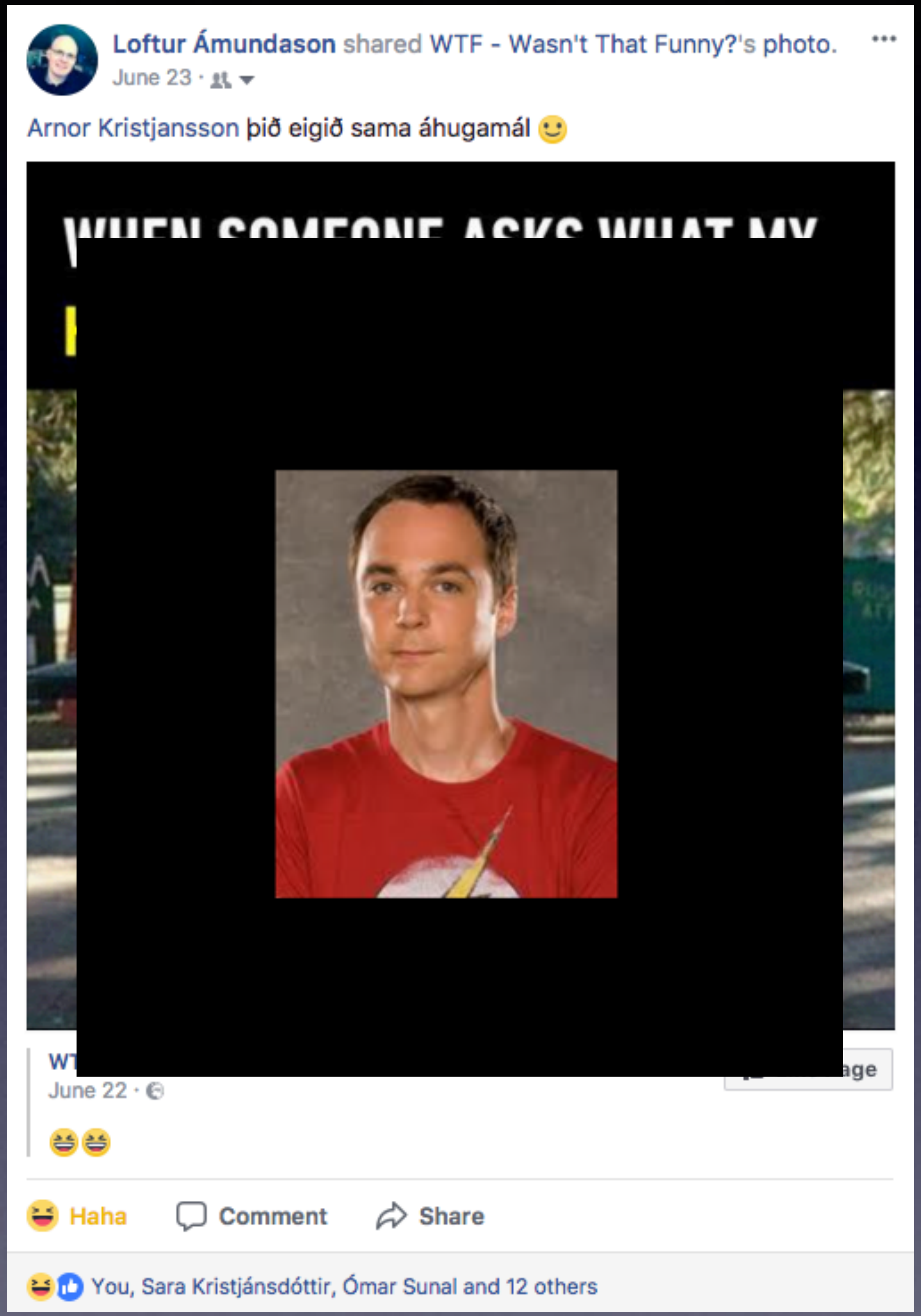
maintain

How to ~~develop for~~ a moving target



Loftur Ámundason shared WTF - Wasn't That Funny?'s photo.

June 23 ·

Arnor Kristjansson þið eigið sama áhugamál 🙂

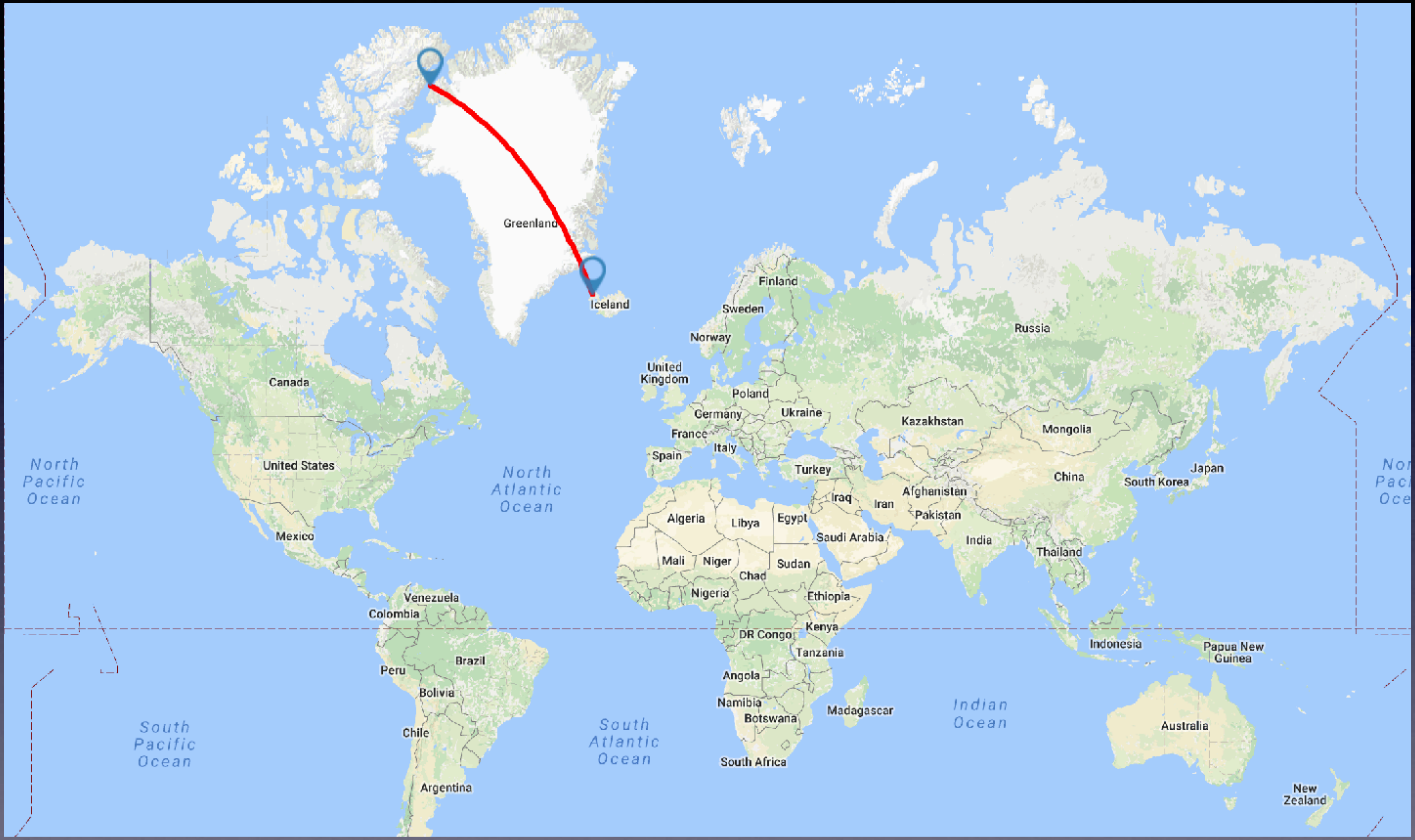
WT

June 22 ·

😆😆

😆👍 You, Sara Kristjánsdóttir, Ómar Sunal and 12 others

70
Total


Greenland
Iceland
Finland
Sweden
Norway
United Kingdom
Poland
Germany
Ukraine
France
Italy
Spain
Turkey
Russia
Kazakhstan
Mongolia
China
Japan
South Korea
Afghanistan
Iraq
Iran
Pakistan
India
Thailand
Egypt
Saudi Arabia
Algeria
Libya
Mali
Niger
Chad
Sudan
Nigeria
Ethiopia
Kenya
DR Congo
Tanzania
Angola
Namibia
Botswana
Madagascar
South Africa
Indonesia
Papua New Guinea
Australia
New Zealand
Canada
United States
Mexico
Venezuela
Colombia
Peru
Brazil
Bolivia
Chile
Argentina
North Pacific Ocean
North Atlantic Ocean
South Pacific Ocean
South Atlantic Ocean
Indian Ocean

Iceland

- “Vikings and sagas”
- “Land of Fire and Ice”
- “Iceland is green and Greenland is ice”

Halldór Guðjónsson
a.k.a. Halldór Kiljan Laxness

- Has been called
  «A scathing poet»
  and «A poet’s poet»

- Won the Nobel Prize in literature in 1955 for the book “Independent People”

- Was very controversial in Iceland, especially since wrote a lot about the idiosyncrasies of Icelanders, and how different layers of society interacted with each other

Heimsljós

- Set in Iceland in the late 1800s
- Published in four parts between 1937 and 1940 and has been called «his most important work»
- English translation published in 1969 as “World Light”

The poem had to

- not contain any obscenity, upon the penalty of death
- be easy to understand
- rhyme
- include her name
- include Jónas' name, the fact that he was a boat-owner and a sheep-farmer who lived at Fótur
- look like Jónas wrote it and should absolutely not be plagiarized from others, like his brother's Júst love poem to Líneik was
- mention that Jónas is the rightful head of the household
- talk badly about Júst, generally and specifically
- include the fact that Líneik is a bloody mare, always biting and kicking and that she'll get plenty of figs and booze, but not in excess – if she behaves

…among other things

My customer story

- the user account should NOT have admin access to the machine
- the Macs to be encrypted (FileVault)
- the user had to be able to unlock the volume on startup
- a separate admin account on the machine, kadmin, had to be able to unlock the volume on startup and be able to log in
- I couldn’t use any form of AD or LDAP, because the management server had to be on the DMZ and those things just weren’t allowed!

…among other things

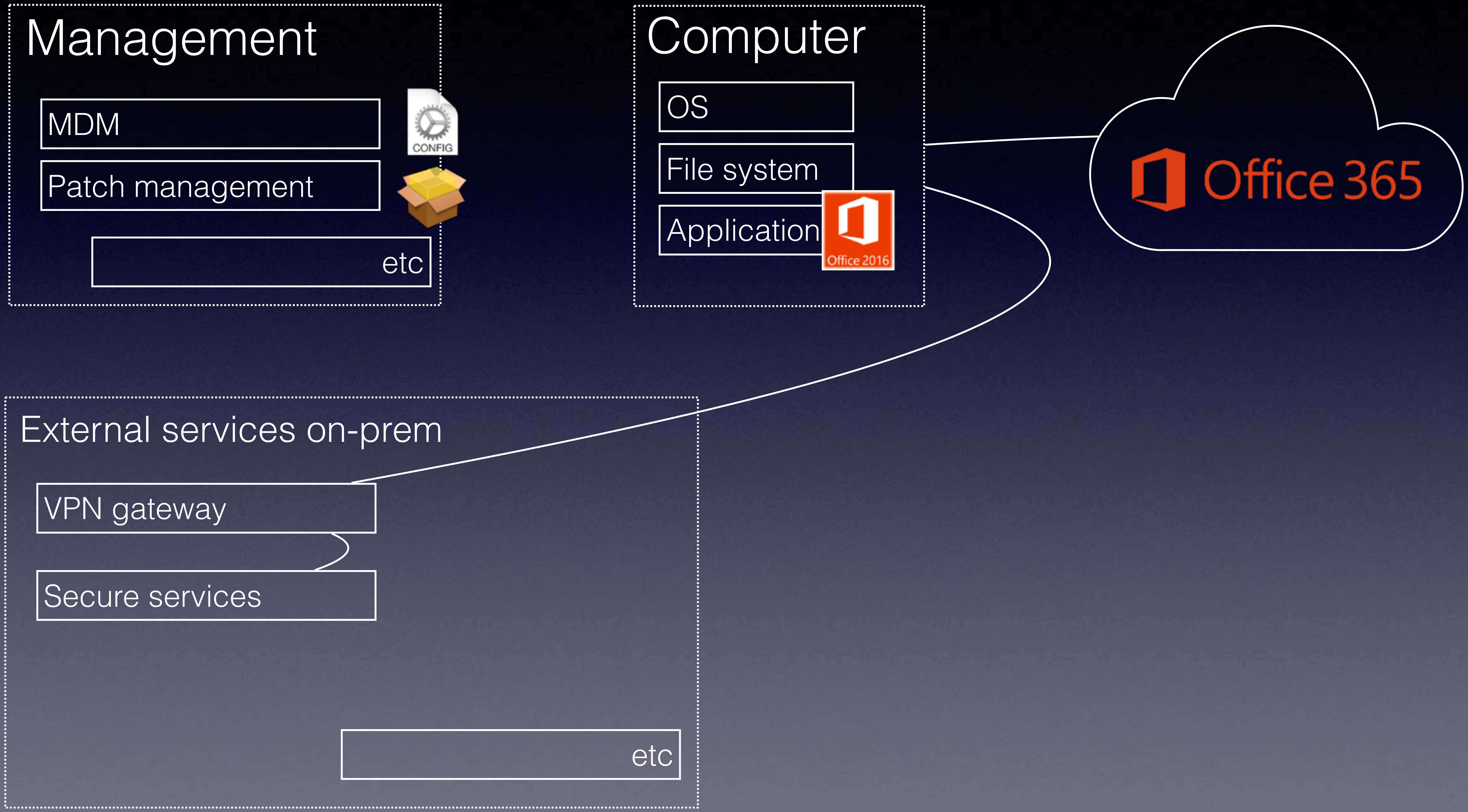
Three types of tools

Tools are not all the same

- There is no “best tool” for Mac management
- All the tools we’re talking about at this conference have their strengths and weaknesses because of the way they’re built

How to choose a tool

- Feature set
- Price
- Availability
- Support
- Community support
- Implementation time
- Complexity
- Fit
- “Single pane of glass”
- Vendor’s reputation
- Ownership requirements
- Documentation



Management

MDM

Patch management

etc

Computer

OS

File system

Application

External services on-prem

VPN gateway

Secure services

etc

Mac management 101

- “Do everything* with an MDM profile”
  - This has limitations, including some technical limitations (depending on the vendor being used)
- Do it manually on each machine
- Write a script that **utilizes** fdesetup, with all it’s multi-user goodness
  - Might be scary to some, but the more you know the better you will become

*everything is a big word. be careful

Using MDM profiles

- MDM payloads are much less approachable than scripts
  - A script is pretty straightforward; if you can get a script to work on a test machine it should work the same when delivered via a management tool
  - .mobileconfig files are 'different'
- MDM profiles cannot do everything (can't install MS Office, Adobe Creative Cloud f.x.)
- No state
- Most vendors (inadvertently) clutter the payloads delivered to the client machines, leading to unforseen consequences and issues, resulting in you having to focus more on the idiosyncrasies of your tool of choice than the task you want to accomplish. For details on this see Erik Nicolas Gomez' April MacBrained presentation: http://bit.ly/2tW8TTa

Do it manually

“This page intentionally left blank”

Use a script

- This will give you 100% control*
- Steeper learning curve
- Fewer bugs**

* if the deployment tool you're using supports the installation and management of scripts on client machines

** you [normally] don't have to deal with the bugs in your deployment tool

Use a mangement tool

- ARD
- munki
- Jamf
- FileWave
- AirWatch
- Meraki
- Parallels
- HEAT LANrev?
- Cortado
- Centrify
- DeployStudio
- Profile Manager

Gordon’s script

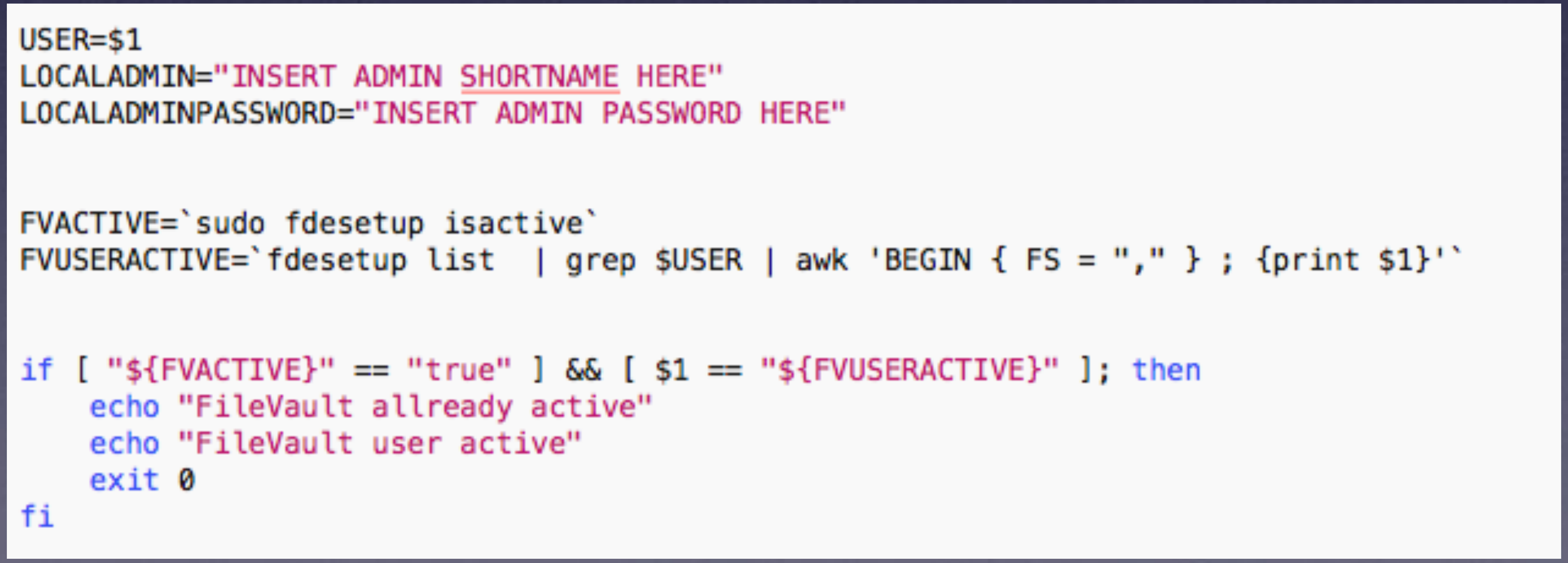
- A script that encrypts the hard disk
- Released when OS X 10.10 was the latest OS
- So let’s disect it…

fdesetup

- sudo fdesetup enable -inputplist < /path/to/fv_users.plist -outputplist > /path/to/recoverykey.plist
- For details on the fv_users.plist, see Rich’s blog post at http://bit.ly/2tOVFUb and Gordon’s 2015 talk at MacSysAdmin

Gordon’s script

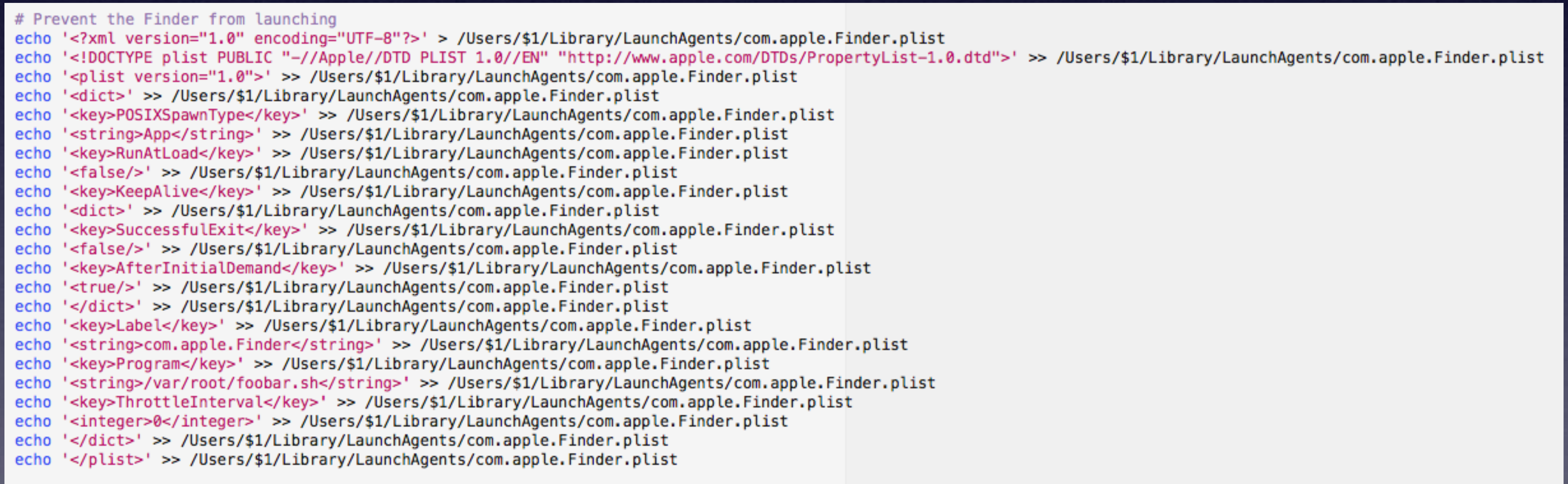
- Gets the initiating user’s username
- Defines the admin username and password
- Checks whether FileVault is enabled and exits if it is



```
USER=$1
LOCALADMIN="INSERT ADMIN SHORTNAME HERE"
LOCALADMINPASSWORD="INSERT ADMIN PASSWORD HERE"


FVACTIVE=`sudo fdesetup isactive`
FVUSERACTIVE=`fdesetup list  | grep $USER | awk 'BEGIN { FS = "," } ; {print $1}'`


if [ "${FVACTIVE}" == "true" ] && [ $1 == "${FVUSERACTIVE}" ]; then
    echo "FileVault allready active"
    echo "FileVault user active"
    exit 0
fi
```

Gordon’s script

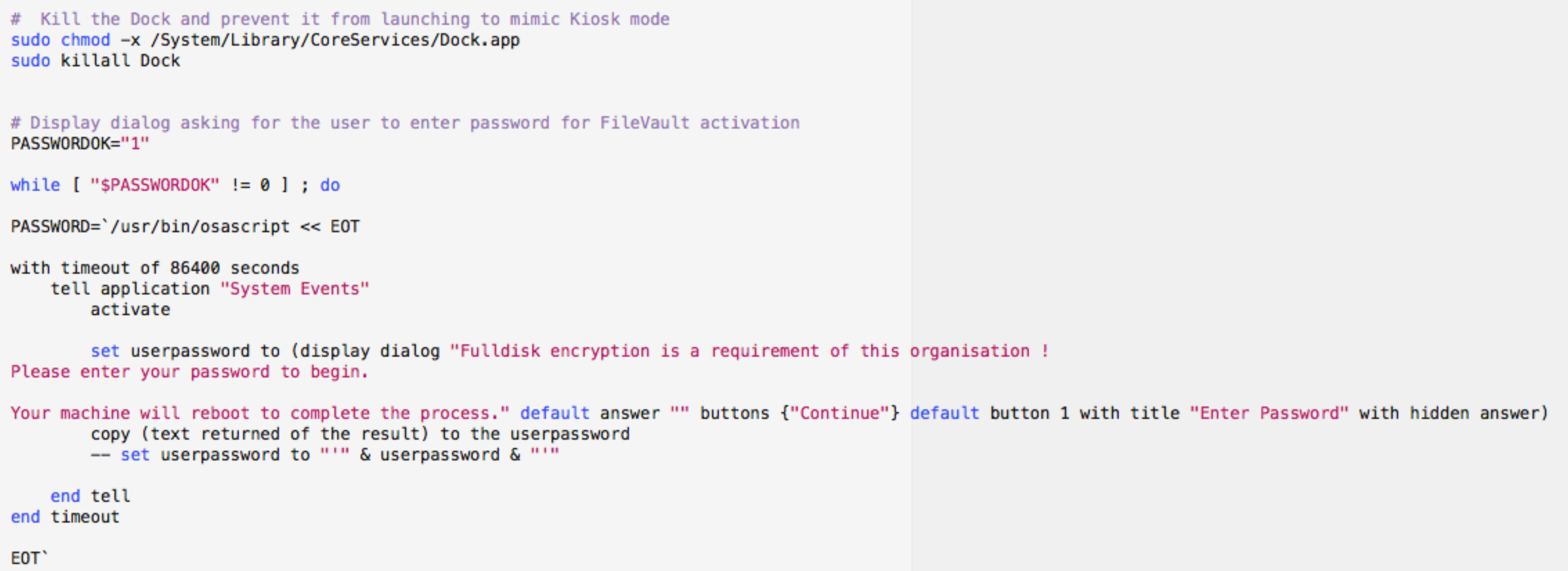
- Disables the Finder



```
# Prevent the Finder from launching
echo '<?xml version="1.0" encoding="UTF-8"?>' > /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<plist version="1.0">' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<dict>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<key>POSIXSpawnType</key>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<string>App</string>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<key>RunAtLoad</key>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<false/>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<key>KeepAlive</key>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<dict>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<key>SuccessfulExit</key>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<false/>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<key>AfterInitialDemand</key>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<true/>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '</dict>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<key>Label</key>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<string>com.apple.Finder</string>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<key>Program</key>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<string>/var/root/foobar.sh</string>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<key>ThrottleInterval</key>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '<integer>0</integer>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '</dict>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
echo '</plist>' >> /Users/$1/Library/LaunchAgents/com.apple.Finder.plist
```

Gordon’s script

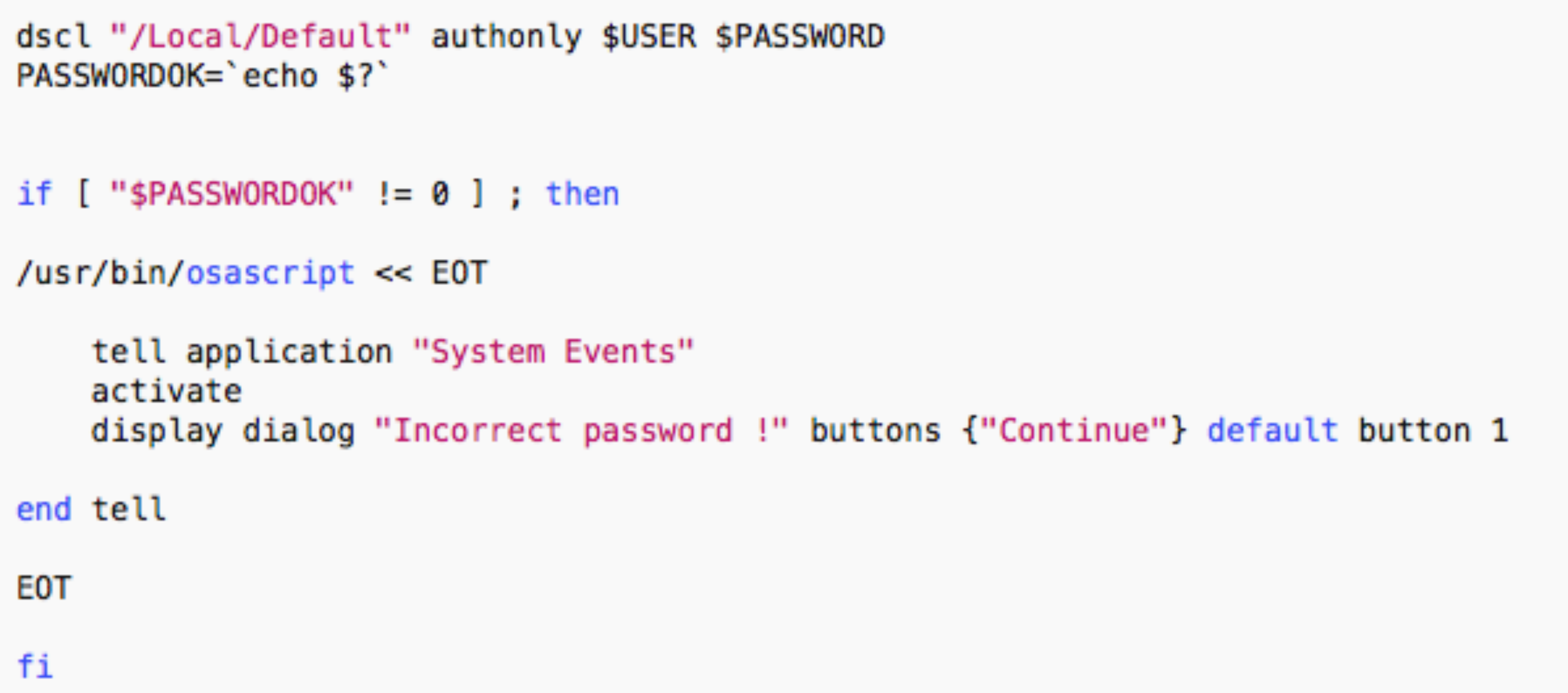
- Kills the dock
- Runs an AppleScript that prompts the user to enter his password



```
#  Kill the Dock and prevent it from launching to mimic Kiosk mode
sudo chmod -x /System/Library/CoreServices/Dock.app
sudo killall Dock


# Display dialog asking for the user to enter password for FileVault activation
PASSWORDOK="1"

while [ "$PASSWORDOK" != 0 ] ; do

PASSWORD=`/usr/bin/osascript << EOT

with timeout of 86400 seconds
    tell application "System Events"
        activate

        set userpassword to (display dialog "Fulldisk encryption is a requirement of this organisation !
Please enter your password to begin.

Your machine will reboot to complete the process." default answer "" buttons {"Continue"} default button 1 with title "Enter Password" with hidden answer)
        copy (text returned of the result) to the userpassword
        -- set userpassword to "'" & userpassword & "'"

    end tell
end timeout

EOT`
```

Gordon’s script

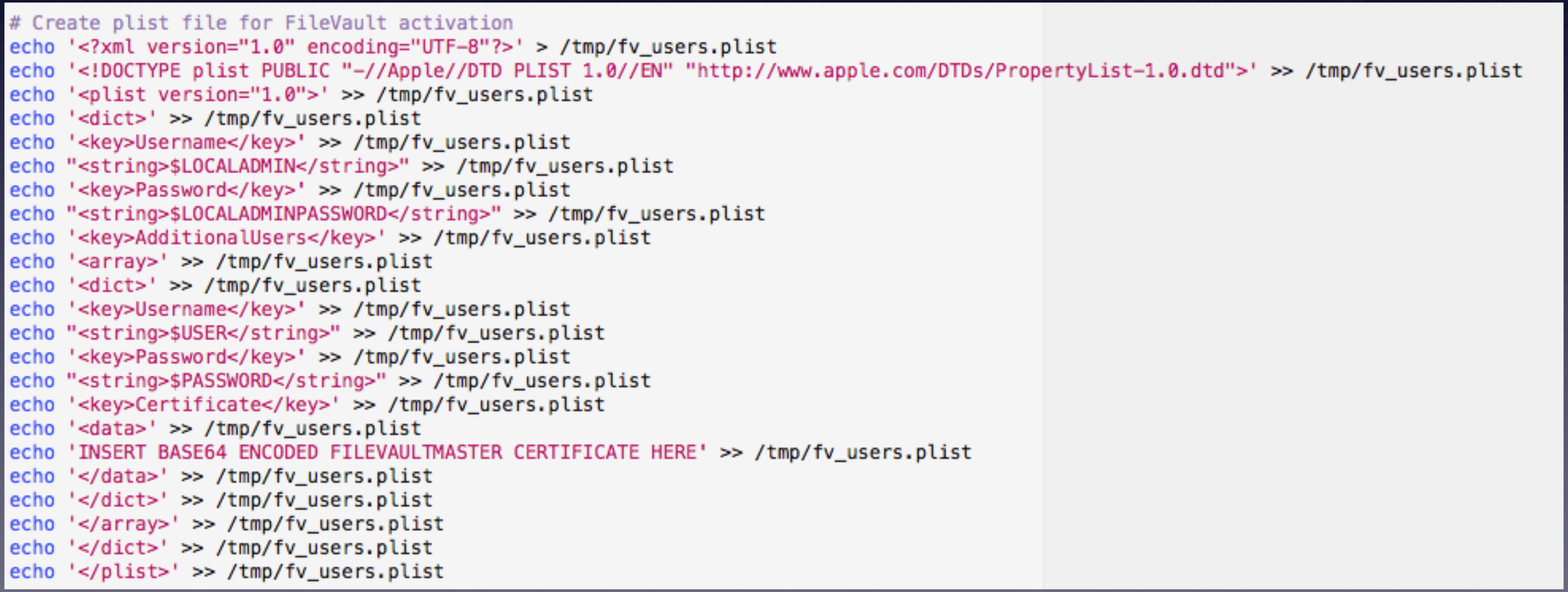
- Checks whether the password is OK and prompts the user to re-enter it, if it is not



```
dscl "/Local/Default" authonly $USER $PASSWORD
PASSWORDOK=`echo $?`


if [ "$PASSWORDOK" != 0 ] ; then

/usr/bin/osascript << EOT

    tell application "System Events"
    activate
    display dialog "Incorrect password !" buttons {"Continue"} default button 1

end tell

EOT

fi
```

Gordon’s script

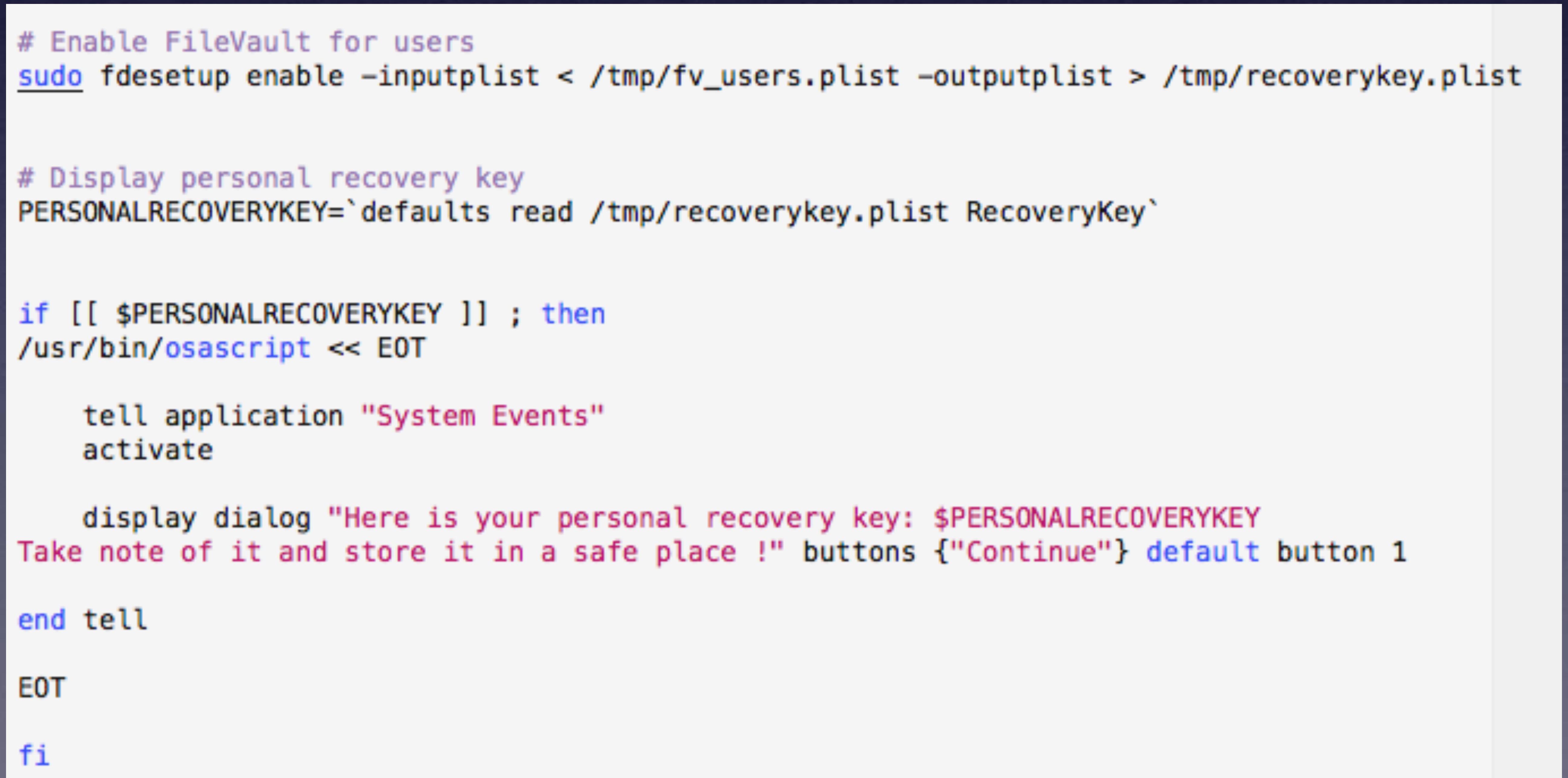
- Creates the fv_users.plist



```
# Create plist file for FileVault activation
echo '<?xml version="1.0" encoding="UTF-8"?>' > /tmp/fv_users.plist
echo '<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">' >> /tmp/fv_users.plist
echo '<plist version="1.0">' >> /tmp/fv_users.plist
echo '<dict>' >> /tmp/fv_users.plist
echo '<key>Username</key>' >> /tmp/fv_users.plist
echo "<string>$LOCALADMIN</string>" >> /tmp/fv_users.plist
echo '<key>Password</key>' >> /tmp/fv_users.plist
echo "<string>$LOCALADMINPASSWORD</string>" >> /tmp/fv_users.plist
echo '<key>AdditionalUsers</key>' >> /tmp/fv_users.plist
echo '<array>' >> /tmp/fv_users.plist
echo '<dict>' >> /tmp/fv_users.plist
echo '<key>Username</key>' >> /tmp/fv_users.plist
echo "<string>$USER</string>" >> /tmp/fv_users.plist
echo '<key>Password</key>' >> /tmp/fv_users.plist
echo "<string>$PASSWORD</string>" >> /tmp/fv_users.plist
echo '<key>Certificate</key>' >> /tmp/fv_users.plist
echo '<data>' >> /tmp/fv_users.plist
echo 'INSERT BASE64 ENCODED FILEVAULTMASTER CERTIFICATE HERE' >> /tmp/fv_users.plist
echo '</data>' >> /tmp/fv_users.plist
echo '</dict>' >> /tmp/fv_users.plist
echo '</array>' >> /tmp/fv_users.plist
echo '</dict>' >> /tmp/fv_users.plist
echo '</plist>' >> /tmp/fv_users.plist
```

Gordon’s script

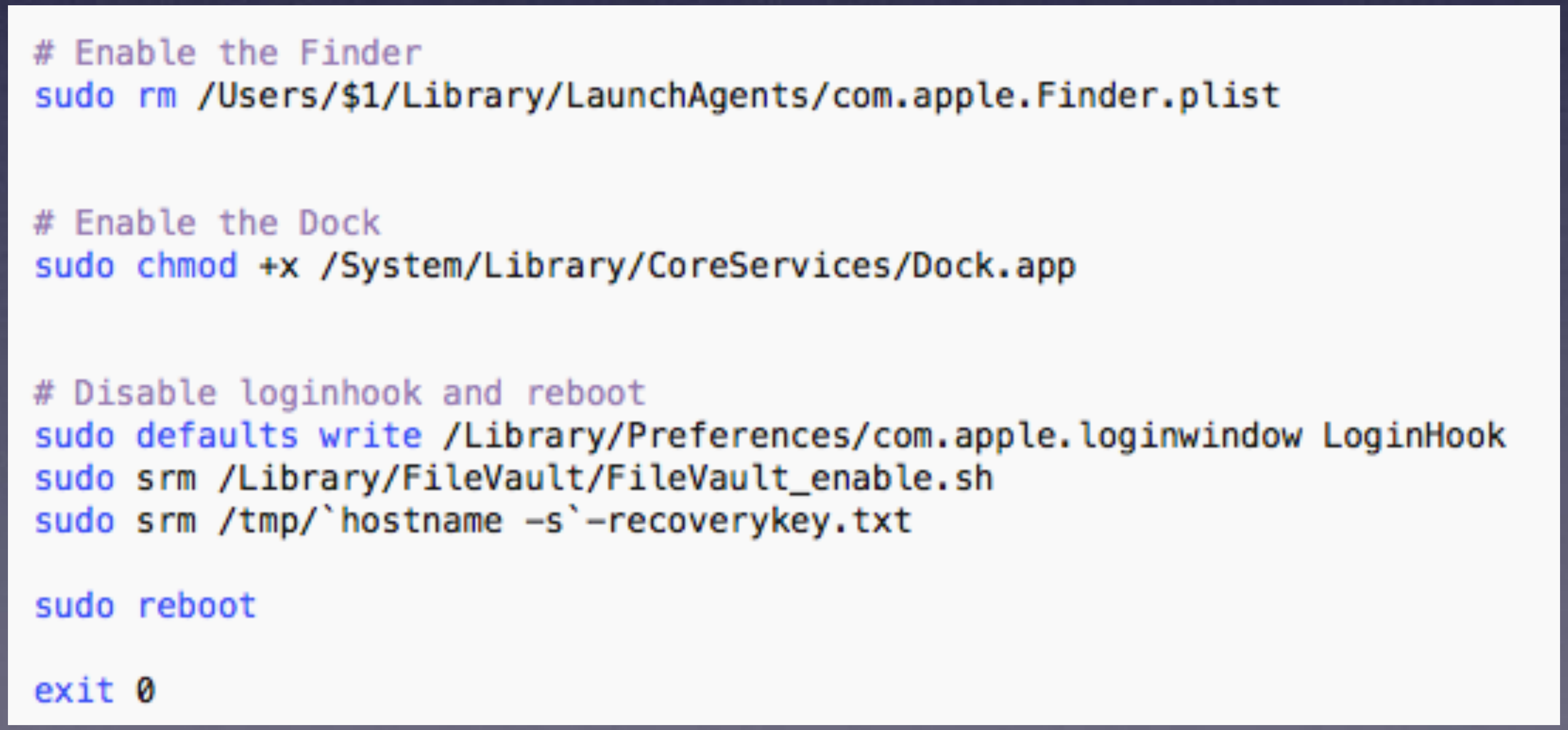
- Runs fdesetup to enable FileVault
- Displays the personal recovery key to the user



```
# Enable FileVault for users
sudo fdesetup enable -inputplist < /tmp/fv_users.plist -outputplist > /tmp/recoverykey.plist


# Display personal recovery key
PERSONALRECOVERYKEY=`defaults read /tmp/recoverykey.plist RecoveryKey`


if [[ $PERSONALRECOVERYKEY ]] ; then
/usr/bin/osascript << EOT

    tell application "System Events"
    activate

    display dialog "Here is your personal recovery key: $PERSONALRECOVERYKEY
Take note of it and store it in a safe place !" buttons {"Continue"} default button 1

end tell

EOT

fi
```

Gordon’s script

- Renables the Dock and Finder
- Disables the LoginHook
- Removes the enablement script and reboots the Mac



```
# Enable the Finder
sudo rm /Users/$1/Library/LaunchAgents/com.apple.Finder.plist


# Enable the Dock
sudo chmod +x /System/Library/CoreServices/Dock.app


# Disable loginhook and reboot
sudo defaults write /Library/Preferences/com.apple.loginwindow LoginHook
sudo srm /Library/FileVault/FileVault_enable.sh
sudo srm /tmp/`hostname -s`-recoverykey.txt

sudo reboot

exit 0
```

Gordon’s script

To deploy the script I needed to

- Generate an admin user using CreateUserPkg
- Put the admin account into the script
- Put the admin password into the script
- Generate a FileVaultMasterKeychain, extract the certificate from the keychain and base64 encode it (base64 /path/to/certificate) and paste the ouput into the correct place in the script

“V2.0”

- Since the customer needed both to encrypt the disk and change the user to a non-admin–both of which require sudo rights–I wrote a LoginHook that
    - checked whether the user logging in was ‘kadmin’ (whether $1=‘kadmin’)
        - if not kadmin, then remove the login user from the admin group using dseditgroup
    - checked whether FileVault was enabled
        - if not, then run the new preparefilevault.sh script
        - if it is, then delete the preparefilevault.sh script
- …and of course created a script, Createhook.sh, that enabled the LoginHook

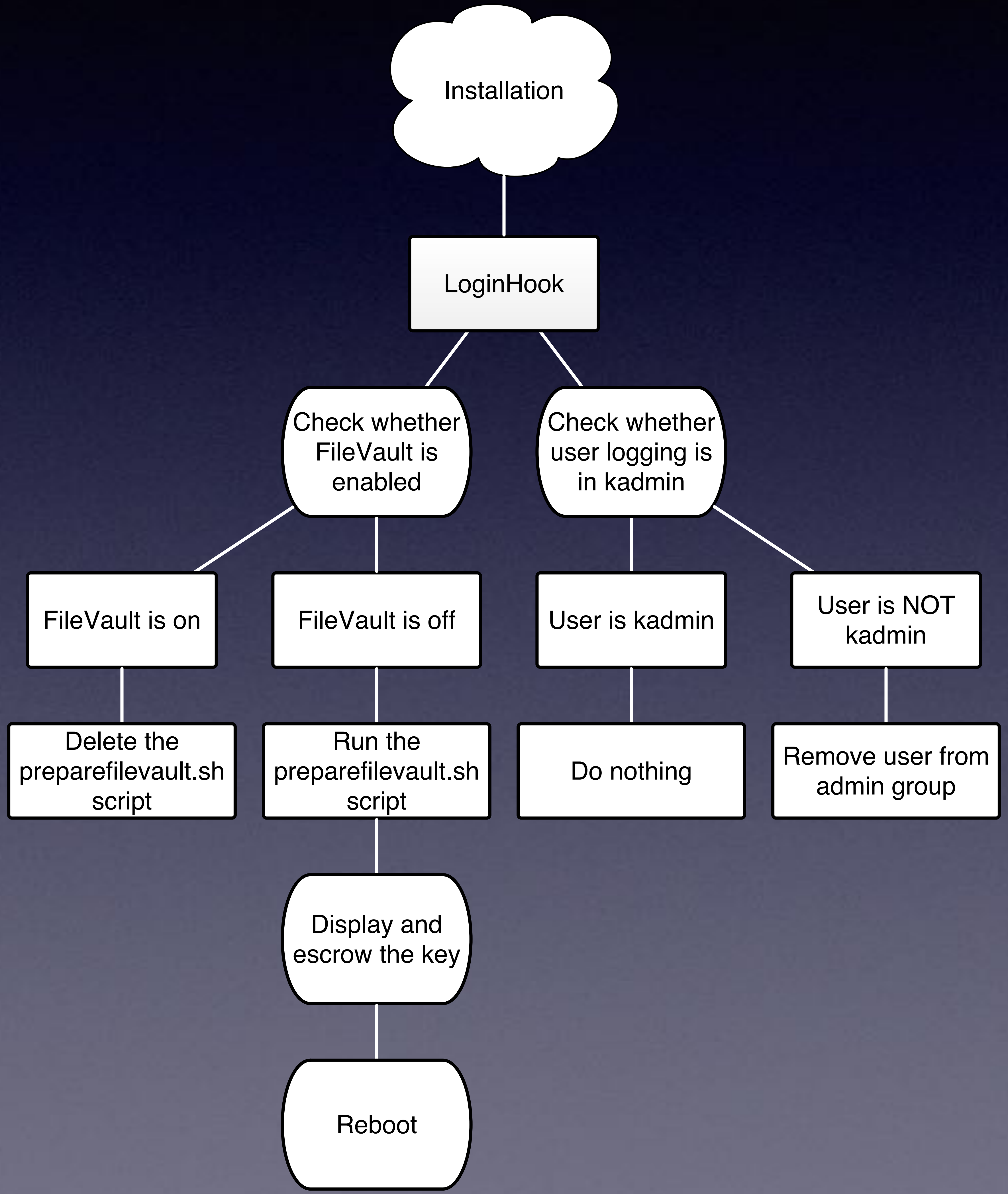
LoginHooks

- Installed by installing a script and then running a “defaults write com.apple.loginwindow LoginHook /path/to/script”
- Runs as **root**
- $1 in the script is used to identify the user logging in

preparefilevault.sh

- Used the AppleScript part of Gordon’s script to prompt the user to enter his password
- Kept the whole “echo > fv_users.plist" part, except for the FileVaultMasterKeychain certificate
  - Opted instead to do a $FILEVAULTCERT variable with the contents of the certificate found in FileVaultMaster.keychain (that was installed as a separate payload)
- escrowed the recovery key to FileWave

The workflow

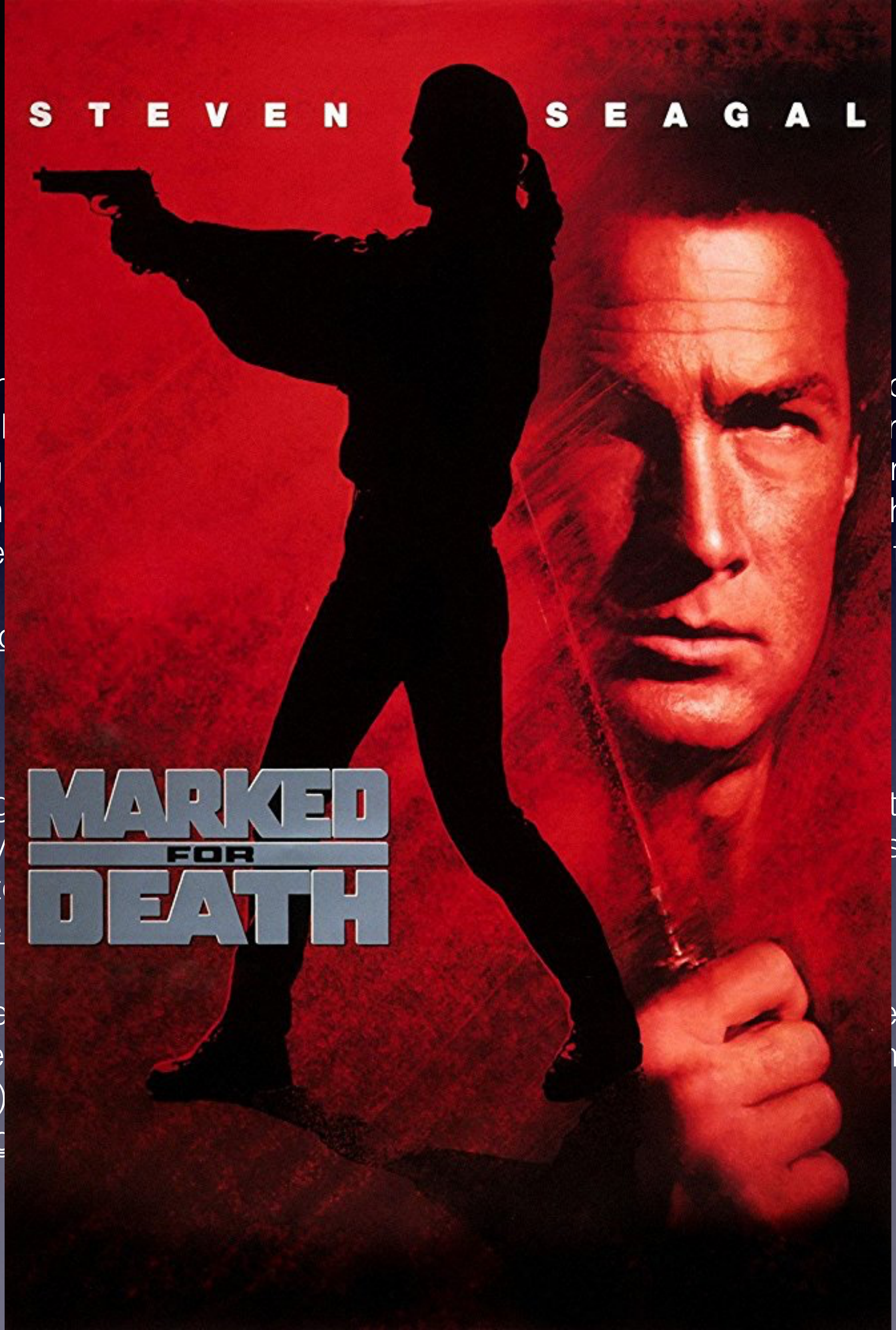


Installation
LoginHook
Check whether FileVault is enabled
Check whether user logging is in kadmin
FileVault is on
FileVault is off
User is kadmin
User is NOT kadmin
Delete the preparefilevault.sh script
Run the preparefilevault.sh script
Do nothing
Remove user from admin group
Display and escrow the key
Reboot

All went well for a while…

Until…

- Time passed; the original FileVaultMaster keychain expired
- OS X 10.11 was released
    - System Image Protection in 10.11 and later doesn’t allow for the script’s method to do kill Dock and hide Finder
- 10.12 was released which *removed* ‘srm’ from the OS, leaving the admin password in cleartext on the machine



“Said of a prograr cess of being
phased out, usual res can,
unfortunately, ling requency in
standards docum hat large amounts
of extant (and pre that have passed
out of favour.”
-www.thefreedictic

“deprecate (v.)
1620s, "to pray ag t participle of
deprecari "to pray ss disapproval" is
from 1640s. Relat
- www.etymonline

“From Latin depre esent or impending
evil), pray for, inte n de (“off”) +
precari (“to pray”)
- en.wiktionary.org

Holy war!

deprecated parts

or “stuff that doesn’t work anymore"

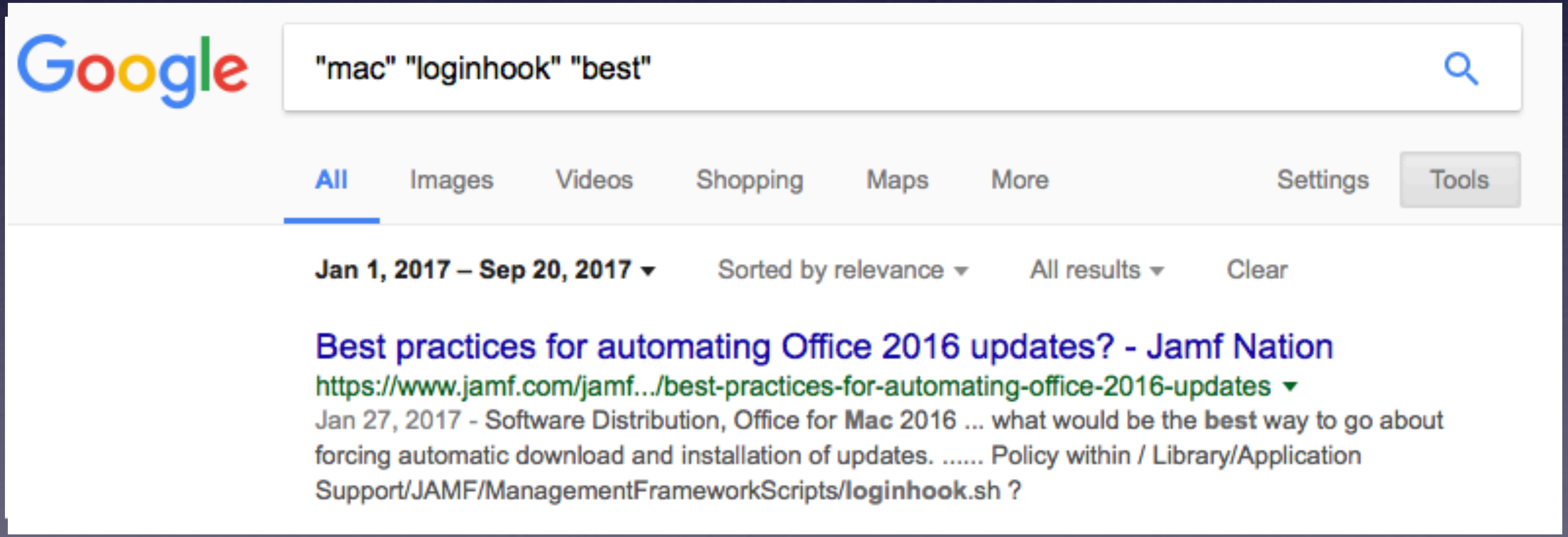
Apple doesn’t document deprecated Terminal commands, only [Cocoa] APIs

- srm - secure remove
  Not deprecated but *removed* from the OS

- LoginHook
  Has been marked as deprecated since OS X Tiger (10.4), released on April 29, 2005 (more than 12 years ago)

LoginHook

- Have been deprecated since OS X 10.4
  - From https://developer.apple.com/library/content/documentation/MacOSX/Conceptual/BPSystemStartup/Chapters/CustomLogin.html:



"mac" "loginhook" "best"

Jan 1, 2017 – Sep 20, 2017 ▾ Sorted by relevance ▾ All results ▾ Clear

Best practices for automating Office 2016 updates? - Jamf Nation
https://www.jamf.com/jamf.../best-practices-for-automating-office-2016-updates ▾
Jan 27, 2017 - Software Distribution, Office for **Mac** 2016 ... what would be the **best** way to go about forcing automatic download and installation of updates. ...... Policy within / Library/Application Support/JAMF/ManagementFrameworkScripts/**loginhook**.sh ?

"But all the cool kids are doing it!"

V3.0

- “Future proof” script – a.k.a. rewrite the script with my angrily new-found knowledge
- Support for multiple languages
- Make it easier to maintain

To accomplish this

- Replace “srm” with “rm” in the script
- The “Kill the Finder and hide the Dock” method didn’t work, because SIP
- FileVaultMasterKeychains are only valid for one year

LaunchAgents and Daemons

to the rescue!

LaunchAgents

- for user tasks; for launching an application at login
- run in the user context; not as root
- can have a GUI

To collect information about the user I used a LaunchAgent to run the AppleScript which has a GUI, that prompts the user to enter his password and to capture the username of the user logging in (using ‘whoami’)

LaunchAgents and Daemons

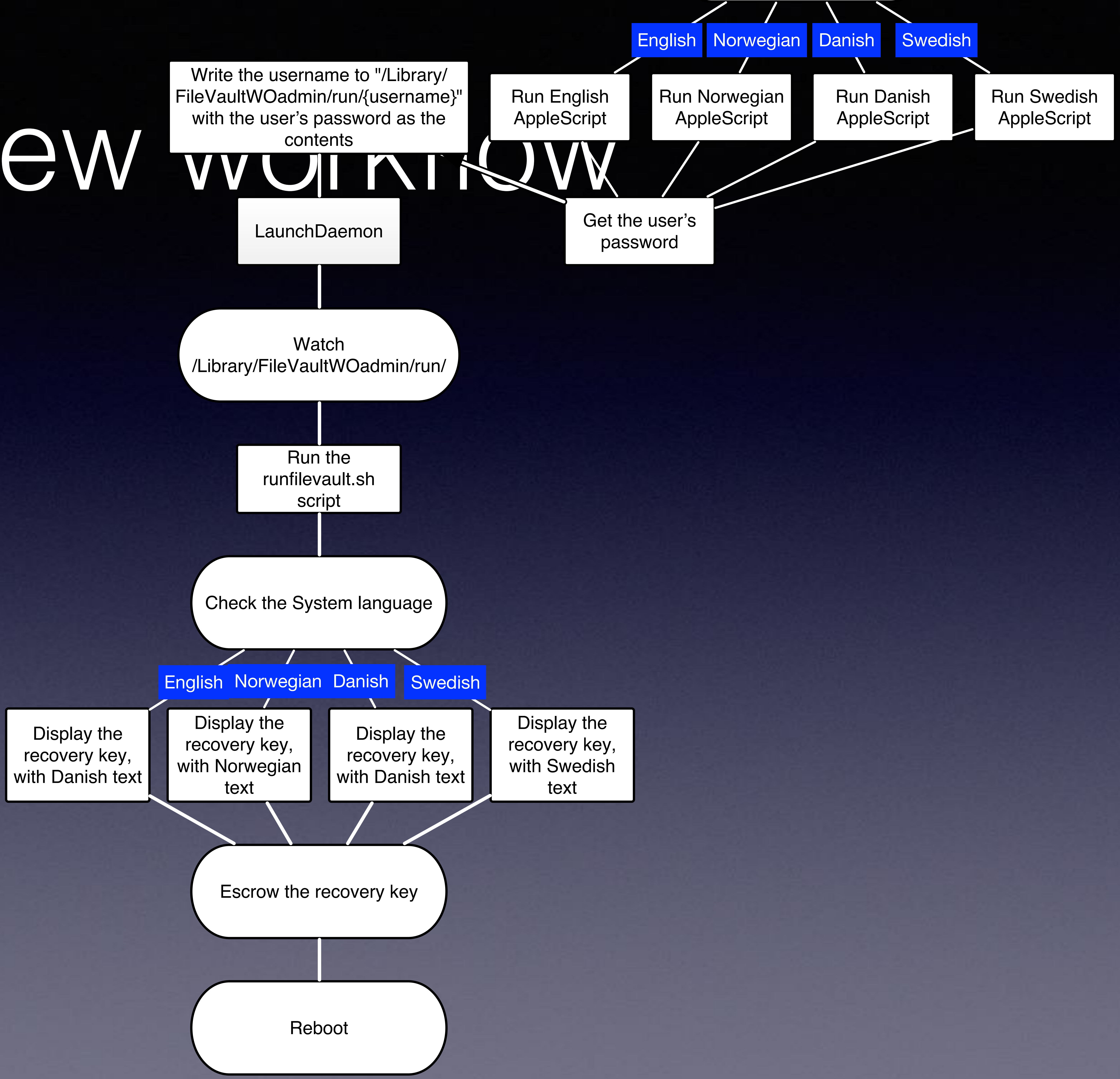
to the rescue!

LaunchDaemons

- are for background processes
- can run as root
- have no idea about the user, because they don’t run in the user’s context
- cannot have a GUI

  Since fdesetup and dseditgroup requires sudo they need to be run in the ‘root’ context; started by a LaunchDaemon (not LaunchAgent)

New Workflow



Get the user's password

English

Norwegian

Danish

Swedish

Run English AppleScript

Run Norwegian AppleScript

Run Danish AppleScript

Run Swedish AppleScript

Write the username to "/Library/FileVaultWOadmin/run/{username}" with the user's password as the contents

LaunchDaemon

Watch /Library/FileVaultWOadmin/run/

Run the runfilevault.sh script

Check the System language

English

Norwegian

Danish

Swedish

Display the recovery key, with Danish text

Display the recovery key, with Norwegian text

Display the recovery key, with Danish text

Display the recovery key, with Swedish text

Escrow the recovery key

Reboot

Demo

The moral of the story

- The environment has a lot to say about what you can and cannot do

- The tool is not the scope; the OS is

- All tools have drawbacks

- Newton’s third law trumps philosophical arguments

- Very few tools help you think about the consequences of your actions

- Ownership trumps technology

The moral of the story

- Don’t think of solutions only in terms of the tools you know/like. Also think about the platform, what it is and where it’s going
- Google only knows what other people know
- The better you understand your* scripts, the more you know
- Experience is the thing of supreme value
  -Henry Ford
- “Assumption is the mother of all fuck-ups”
  -Marcus Penn
- Software is **never** finished

*also applies to scripts written by others

"I know a girl who far outshines all others,
A lively filly in a herd of mothers;
With lovely eyes and legs so neat,
She neighs and bites and kicks her feet,
Oh, maiden sweet!
Sweet buds of Christian love will flower
Around her bower.

They've all been trying hard, I know, to win her,
But none has managed, yet I think, to pin her;
Until she found a man to keep,
A boat-owner who handles sheep,
Both wise and deep.
Sweet buds of Christian love will flower
Around her bower.

He'll give her brennivín in moderation,
Sweets and figs and fruit from every nation;
If she doesn't lose her charm,
He will give her Fótur, his farm,
And his strong arm.
Sweet buds of Christian love will flower
Around her bower."

Thank you