

TWISTING MUNKI

(for configuration management)

@GRAHAMGILBERT

pebble.it



MUNKI ISN'T A
CONFIGURATION
MANAGEMENT TOOL

But it can pretend to be one

**"I DON'T USE
MUNK!"**

GRAHAM'S FIRST LAW OF MAC MANAGEMENT

**CONFIGURATION NEEDS
TO BE CONTINUALLY
MANAGED**



Here be dragons.

A photograph showing a row of terraced houses made of light-colored stone or brick. The houses have dark roofs and multiple windows. In the foreground, there's some greenery and a paved path. The sky is overcast.

CURRENT STATE

A scenic tropical beach with clear blue water, white sand, and lush green limestone cliffs in the background.

DESIRED STATE

A teal-colored lunchbox with a red handle and a red star sticker on the front. The word "CORRECTION" is overlaid in large, bold, red letters.

CORRECTION

**HOW MUNKI DECIDES
WHAT NEEDS TO
BE INSTALLED**

**INSTALL CHECK SCRIPT
INSTALLS ARRAY
RECEIPTS**

**INSTALL CHECK SCRIPT
INSTALLS ARRAY
RECEIPTS**

```
$ makepkginfo -f /Library/Management/Scripts/wallpaper
```

```
<key>installs</key>
<array>
    <dict>
        <key>md5checksum</key>
        <string>85757f9a341df4de3f1ca01b6a1d52eb</string>
        <key>path</key>
        <string>/Library/Management/Scripts/wallpaper</string>
        <key>type</key>
        <string>file</string>
    </dict>
</array>
```

GRAHAM'S SECOND LAW OF MAC MANAGEMENT

USERS ** LOVE TO
CLICK ‘LATER’**

**UNATTENDED
INSTALL**

**FORCED INSTALL
AFTER DATE**

I'M UPDATING!



INSTALL CHECK SCRIPT
INSTALLS ARRAY
RECEIPTS

Blah blah blah

TIME SERVER

```
#!/bin/bash  
  
/usr/sbin/systemsetup -setnetworktimeserver ntp.company.com  
  
/usr/sbin/systemsetup -setusingnetworktime on
```

```
USE_PKGBUILD=1
include /usr/local/share/luggage/luggage.make

TITLE=set_time_server
PACKAGE_NAME=${TITLE}
REVERSE_DOMAIN=com.pebbleit
PAYLOAD=\
    pack-script-postinstall
```



Date & Time

Search

Date & Time

Time Zone

Clock



Set date and time automatically:

ntp.company.com

29/09/2014

17:09:40



To set date and time formats, use Language & Region preferences.

[Open Language & Region...](#)

```
$ pkgutil --pkg-info com.pebbleit.set-time-server  
package-id: com.pebbleit.set-time-server  
version: 1.0.0  
volume: /  
location: /  
install-time: 1411314264
```



Date & Time

Search

Date & Time

Time Zone

Clock

 Set date and time automatically: Apple Europe (time.euro.apple.com.)

29/09/2014

17:21:47



To set date and time formats, use Language & Region preferences. [Open Language & Region...](#)





2. grahamgilbert@Artoo: ~ (zsh)

Last login: Mon Sep 29 16:18:24 on ttys000

[grahamgilbert@Artoo] []

└▷ sudo managedsoftwareupdate [~]

Password:

Managed Software Update Tool

Copyright 2010-2014 The Munki Project

<http://code.google.com/p/munki>

Starting...

Checking for available updates...

 Retrieving list of software for this machine...

 Retrieving list of software for this machine...

Finishing...

 Performing postflight tasks...

 postflight stdout: Sal report submitted for Artoo

.

Done.

[grahamgilbert@Artoo] []

└▷ [~]

**INSTALL
CHECK
SCRIPTS**

exit 1
EVERYTHING'S PEACHY

exitⁱ0
I NEED TO BE

```
#!/bin/bash

TIME SERVER=`/usr/sbin/systemsetup -getnetworktimeserver`  
USING NETWORK=`/usr/sbin/systemsetup -getusingnetworktime`  
  
if [ "$TIME SERVER" == "Network Time Server: ntp.company.com" ]; then  
    if [ "$USING NETWORK" == "Network Time: On" ]; then  
        exit 1  
    else  
        exit 0  
    fi  
else  
    exit 0  
fi
```

security



security(1)

BSD General Commands Manual

security(1)

NAME**security** -- Command line interface to keychains and Security framework**SYNOPSIS****security** [-hilqv] [-p prompt] [command] [command_options] [command_args]**DESCRIPTION**

A simple command line interface which lets you administer keychains, manipulate keys and certificates, and do just about anything the Security framework is capable of from the command line.

By default **security** will execute the command supplied and report if anything went wrong.

If the **-i** or **-p** options are provided, **security** will enter interactive mode and allow the user to enter multiple commands on stdin. When EOF is read from stdin **security** will exit.

Here is a complete list of the options available:

-h If no arguments are specified, show a list of all commands. If



```
Last login: Mon Sep 29 17:31:35 on ttys002
[grahamgilbert@Artoo] [ ]
↳ security authorizationdb read system.preferences.energysaver          [~]
<?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>allow-root</key>
    <true/>
    <key>authenticate-user</key>
    <true/>
    <key>class</key>
    <string>user</string>
    <key>comment</key>
    <string>Checked by the Admin framework when making changes to the Energy Saver preference pane.</string>
    <key>created</key>
    <real>427674229.99584597</real>
    <key>group</key>
    <string>admin</string>
    <key>modified</key>
    <real>427674229.99584597</real>
    <key>session-owner</key>
    <false/>
    <key>shared</key>
    <true/>
    <key>timeout</key>
    <integer>2147483647</integer>
    <key>tries</key>
    <integer>10000</integer>
    <key>version</key>
    <integer>0</integer>
</dict>
</plist>
YES (0)
[grahamgilbert@Artoo] [ ]
```

```
#!/usr/bin/env python

import subprocess
import sys
import plistlib

# Group System Preferences should be opened to
group = 'everyone'

command = ['/usr/bin/security', 'authorizationdb', 'read', 'system.preferences']

task = subprocess.Popen(command, stdout=subprocess.PIPE, stderr=subprocess.PIPE)
(out, err) = task.communicate()

formatted = plistlib.readPlistFromString(out)
# if group matches, exit 1 as we don't need to install
if formatted['group'] == group:
    sys.exit(1)
else:
    # if it doesn't we're exiting with 0 as we need to perform the install
    sys.exit(0)
```

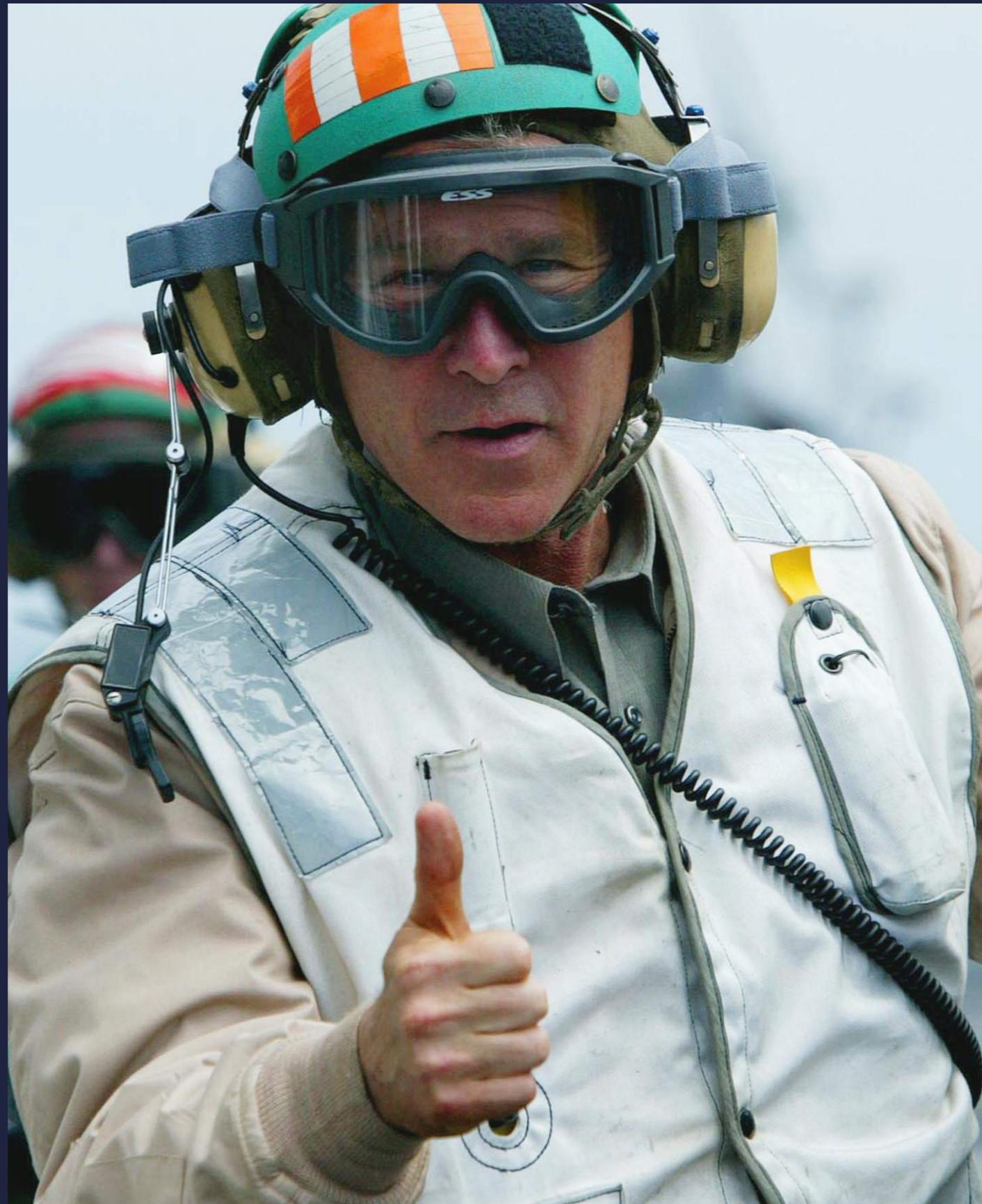
PROFILES

```
/usr/bin/profiles -I -F /path/to/my.mobileconfig
```

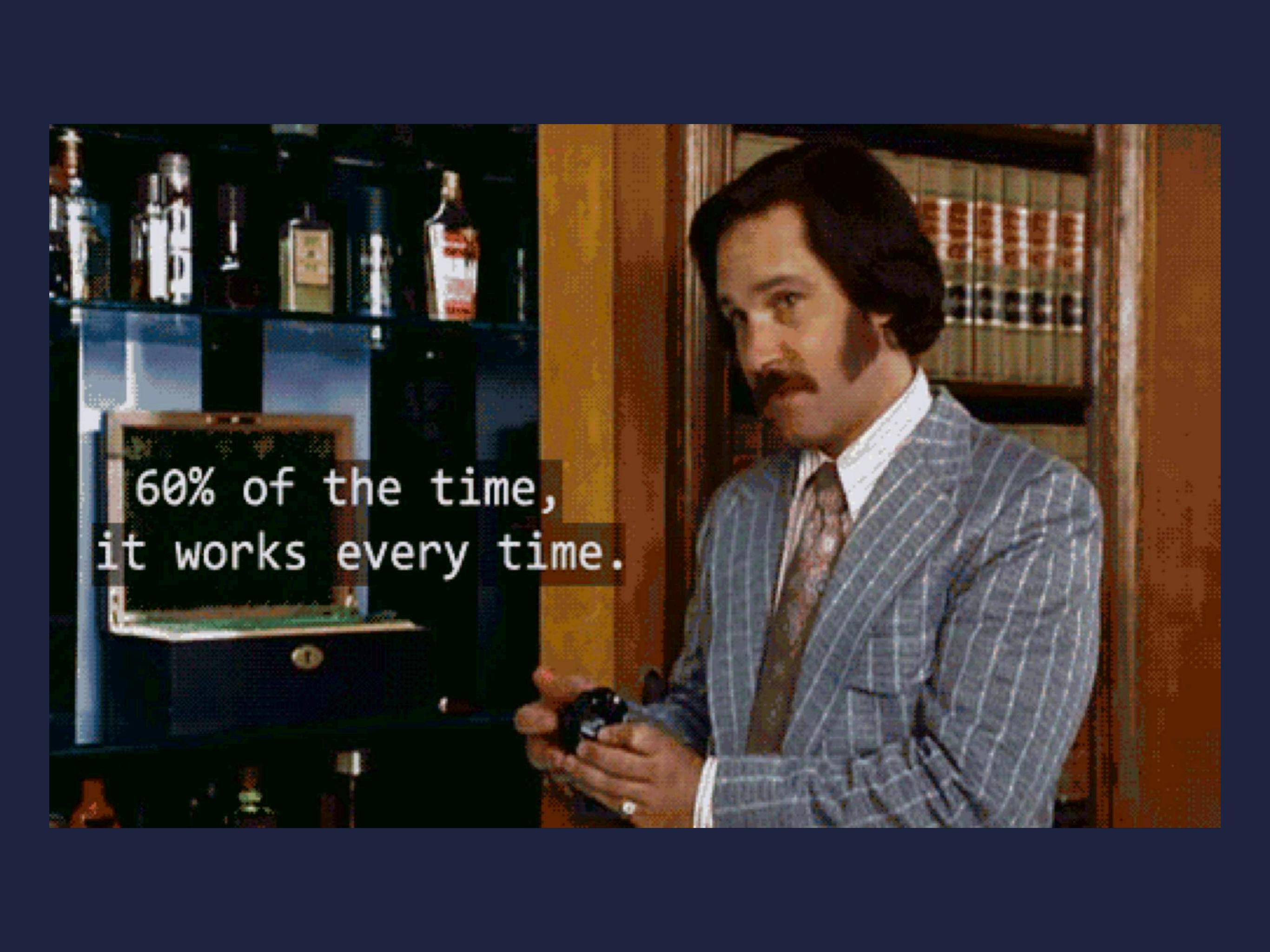
MAKE-PROFILE-PKG

```
#!/bin/bash

if ( /usr/bin/profiles -P | /usr/bin/grep -q 'com.pebbleit.myprofile' ); then
    # Profile is present
    exit 1
else
    # Profile isn't there, need to install
    exit 0
fi
```



BUT WAIT!

A color photograph of a man with dark hair and a well-groomed mustache, wearing a light-colored pinstripe suit jacket over a white shirt and patterned tie. He is standing in a bar setting, looking directly at the camera with a slight smile. In his right hand, he holds a dark bottle, likely Jim Beam Kentucky Straight Bourbon Whiskey, with a prominent label. To his left, a dark wooden bar counter is visible, featuring several bottles of alcohol on shelves and a small framed picture or mirror on the wall above it.

60% of the time,
it works every time.

**INSTALL CHECK SCRIPT
INSTALLS ARRAY
RECEIPTS**

INSTALL CHECK SCRIPT
INSTALLS ARRAY
RECEIPTS

```
#!/bin/bash

# The version of the package
PKG_VERSION="1.0.0"

# The identifier of the package
PKG_ID="com.pebbleit.myprofile"

# The identifier of the profile
PROFILE_ID="com.pebbleit.myprofile"

# The version installed from pkgutil
VERSION_INSTALLED=`/usr/sbin/pkgutil --pkg-info ${PKG_ID} | grep version | sed 's/^[:^]*://'`

if ( /usr/bin/profiles -P | /usr/bin/grep -q $PROFILE_ID ); then
    # Profile is present, check the version
    if [ "$VERSION_INSTALLED" = "$PKG_VERSION" ]; then
        # Correct version, all good
        exit 1
    else
        exit 0
    fi
else
    # Profile isn't there, need to install
    exit 0
fi
```

**WHO LOVES
PRINTERS?**

space:		MSU End:	2014-09-29
Console user:	grahamgilbert		20:51
		Puppet Run:	2014-09-29
			20:36

Errors 9

Running postinstall_script for laserjet_p4014 failed.

```
-----  
  (standard_in) 1: illegal character &  
  (standard_in) 1: parser error at or before  
printer laserjet_p4014: id . enabled in M Z step 20:49:35 2014  
/tmp/munki-SPyeoQ/postinstall_script: line 3: [ == binary operator expected  
lpadmin: Unable to open PPD file '/Library/Printers/Postscript/Contents/Resources/HP Las  
erJet P4010_P4510 Series.gz" - No such file or directory  
  File Doesn't Exist, Will Create: /private/etc/cups/deployment/receipts/laserjet_p4  
014.plist  
-----
```

Warnings 1

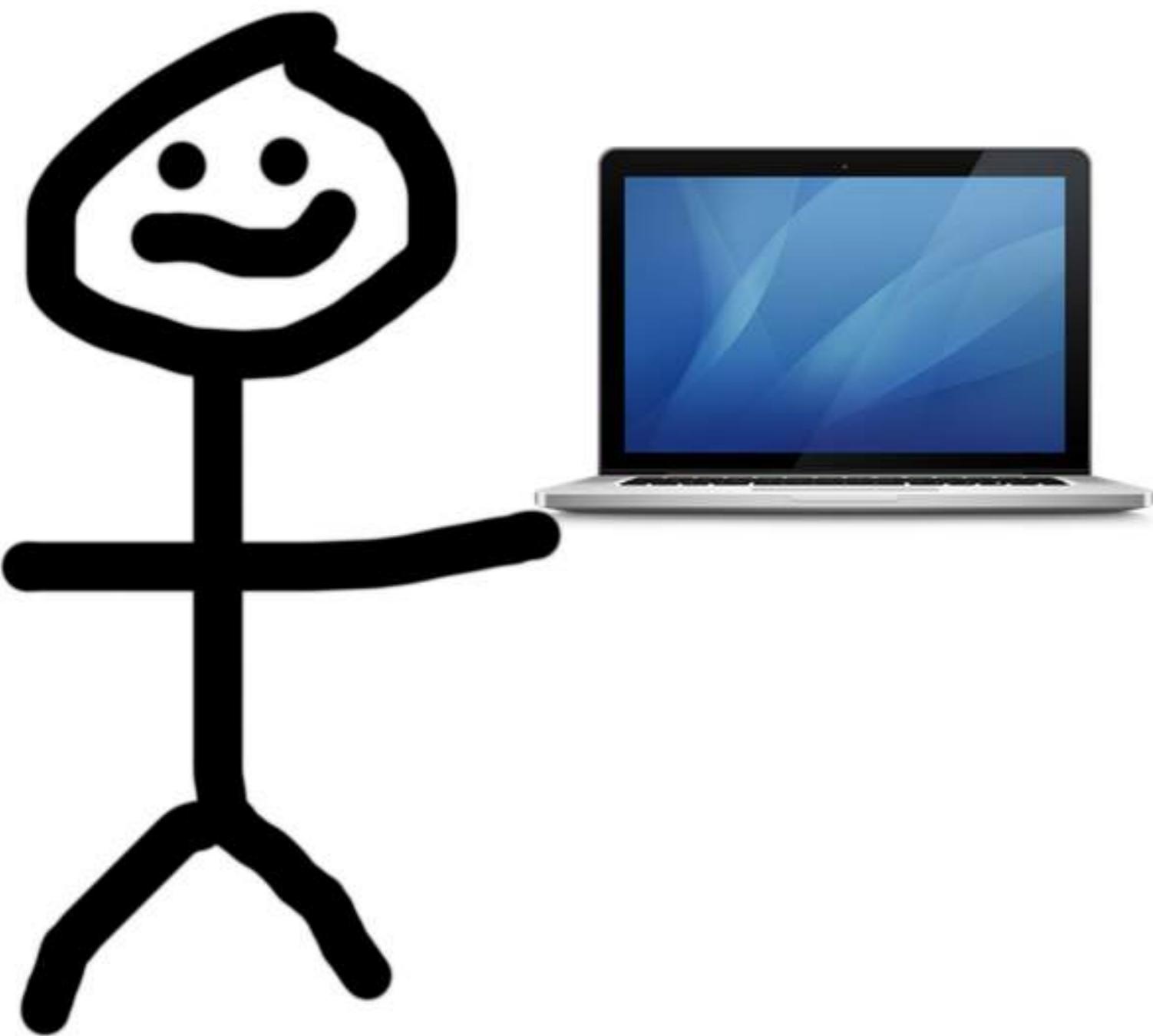
Postinstall script for laserjet_p4014 returned 1

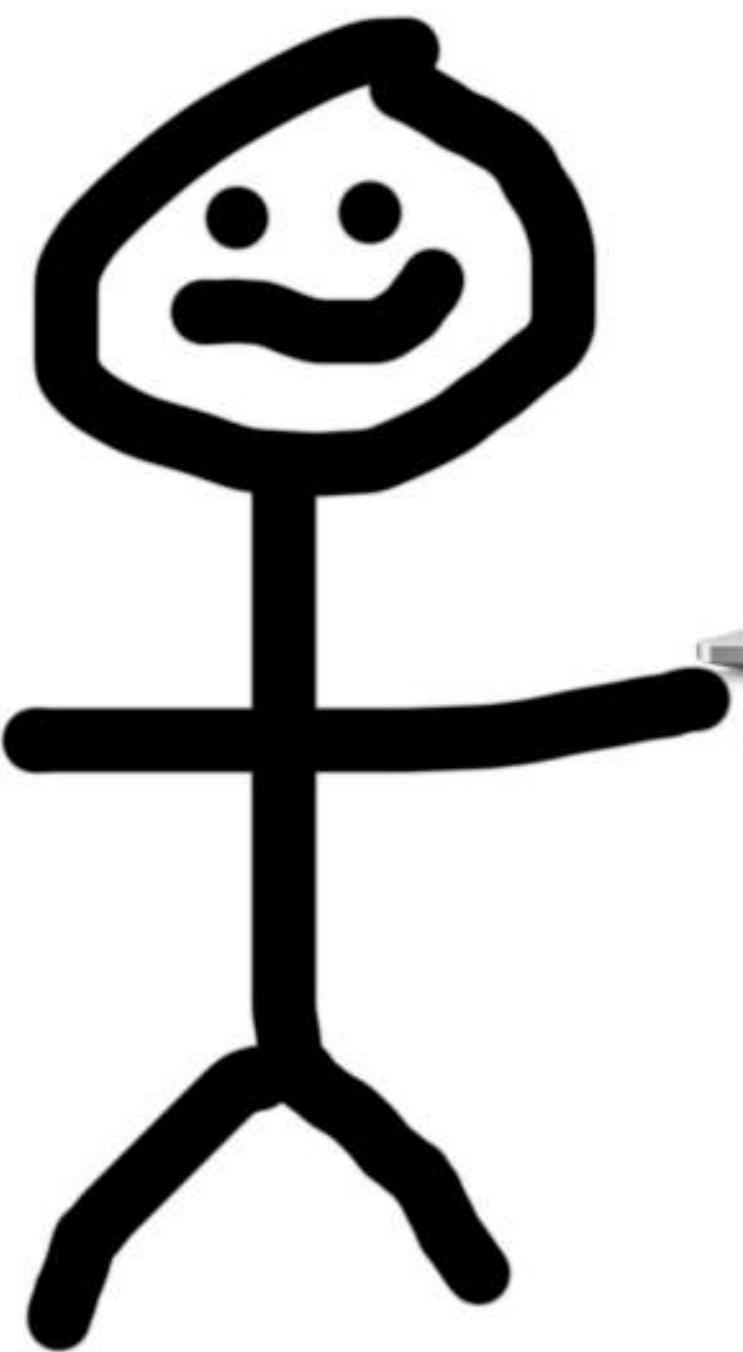
RELATIONSHIPS

```
<key>catalogs</key>
<array>
    <string>production</string>
</array>
<key>category</key>
<string>Printers</string>
<key>description</key>
<string>Sets up the HP LaserJet P4014</string>
<key>display_name</key>
<string>HP LaserJet P4014</string>
<key>name</key>
<string>laserject_p4014</string>
<key>requires</key>
<array>
    <string>HewlettPackardPrinterDrivers</string>
</array>
```

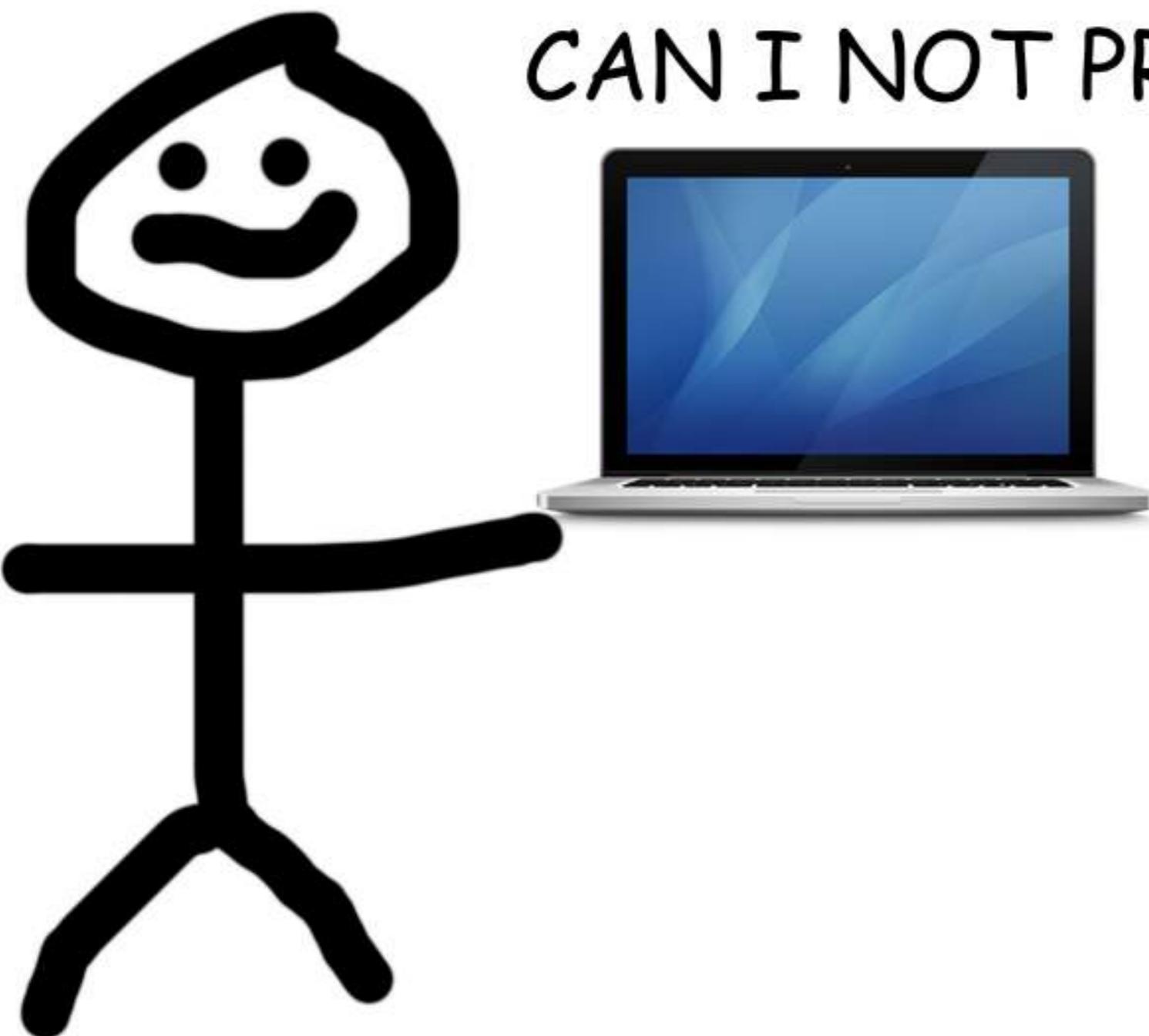
```
<key>requires</key>
<array>
    <string>HewlettPackardPrinterDrivers</string>
</array>
```

CONDITIONAL ITEMS





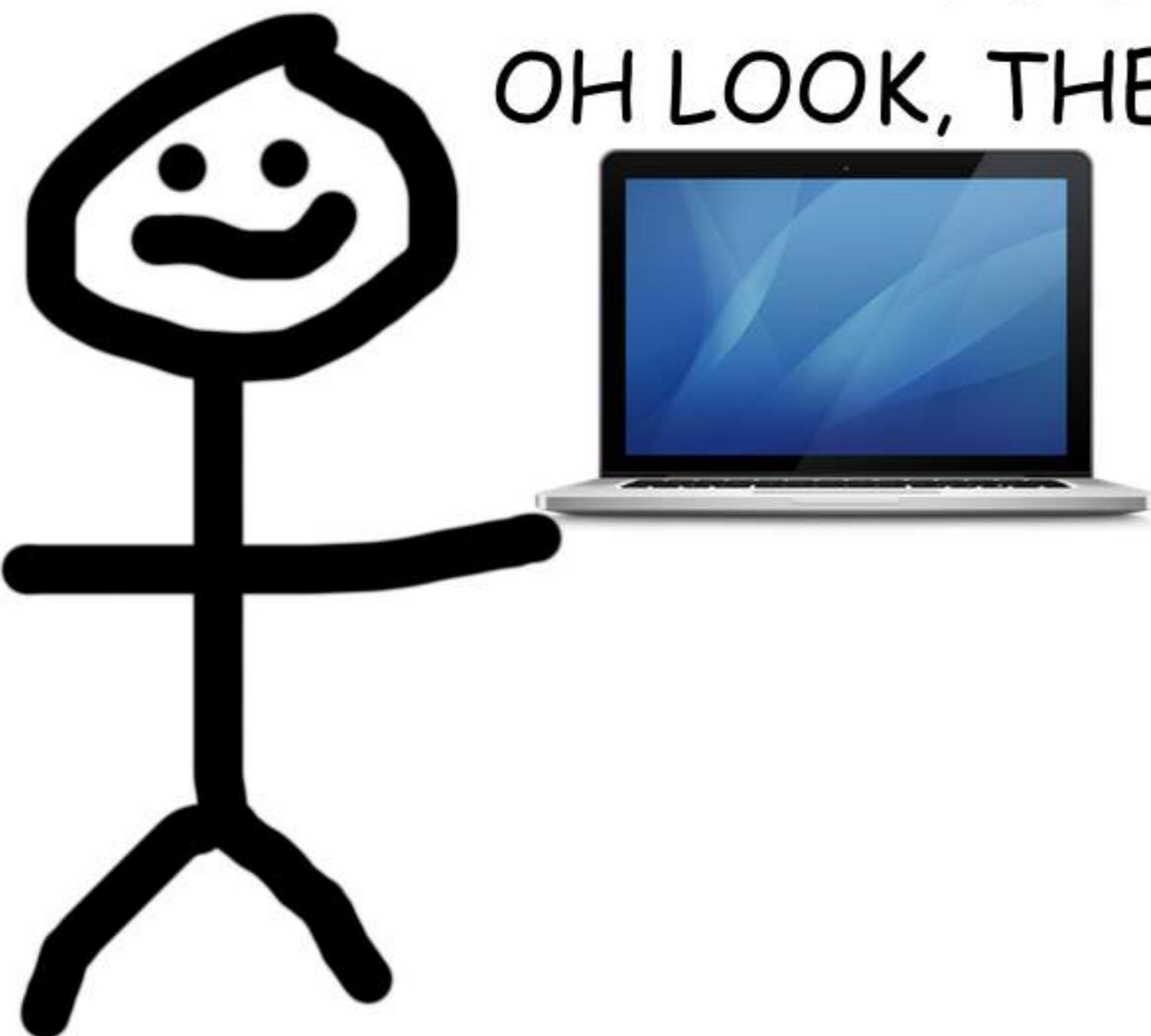
I NEED TO PRINT MY VERY
IMPORTANT DOCUMENT. WHY
CAN I NOT PRINT! MAKE IT
PRINT.





```
<key>conditional_items</key>
<array>
    <dict>
        <key>condition</key>
        <string>external_ip == "123.456.789.123"</string>
        <key>managed_installs</key>
        <array>
            <string>InstallLondonPrinter</string>
        </array>
    </dict>
</array>
```

I NEED TO PRINT MY VERY
IMPORTANT DOCUMENT.
OH LOOK, THERE'S THE PRINTER.



SELF SERVICE

OPTIONAL
INSTALS



SELFSERVEMANIFEST

```
<?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>managed_installs</key>
  <array>
    <string>OracleJava8</string>
    <string>Silverlight</string>
  </array>
  <key>managed_uninstalls</key>
  <array>
  </array>
</dict>
</plist>
```

```
#!/usr/bin/python

import plistlib

my_item = 'RepairPermissions'

SelfServeManifestPath = "/Library/Managed Installs/manifests/SelfServeManifest"
SelfServeManifest = plistlib.readPlist(SelfServeManifestPath)
for item in SelfServeManifest.get('managed_installs', []):
    if item == my_item:
        SelfServeManifest['managed_installs'].remove(item)

SelfServeManifest = plistlib.writePlist(SelfServeManifest, SelfServeManifestPath)
```

INSTALLINFO

I THOUGHT WE WERE REPAIRING PERMISSIONS?

```
#!/usr/bin/python

import subprocess

# Run the repair permissions

cmd = ['/usr/sbin/diskutil', 'repairPermissions', '/']
task = subprocess.Popen(cmd).communicate()[0]
```

PREFLIGHT
POSTFLIGHT

WHERE AM I?

PACKAGEUR

NO INTERRUPTIONS



Hello, IT, have you tried
turning it off and on again?

```
#!/bin/bash

days=`uptime | awk '{ print $4 }' | sed 's/,//g'`
num=`uptime | awk '{ print $3 }'`

if [ $days = "days" ]; then
    if [ $num -gt "7" ]; then
        exit 0
    else
        exit 1
    fi
else
    exit 1
fi
```

Managed Software Centre

< | >

Software Categories My Items Updates

Search

1 pending update **Restart Required**

System Reboot Required.
Version 1.0
Restart Required ⏪

This will be enforced after 14 days.

Software | Categories | Updates

```
<key>forced_install_after_date</key>
<date>2011-08-03T13:00:00Z</date>
```

EVERYONE NEEDS A NAME

CSV Package

Requires

Is the Mac's
name the same
as the CSV?

No

Proceed with
install

Yes

Exit

Exit

```
#/usr/bin/python
```

```
import blah
```

```
if blah != blahblah:  
    print "Time for tea, chaps"
```

WHAT'S MISSING?

REPORTING

Clients

396
Clients**208**
Req/hour

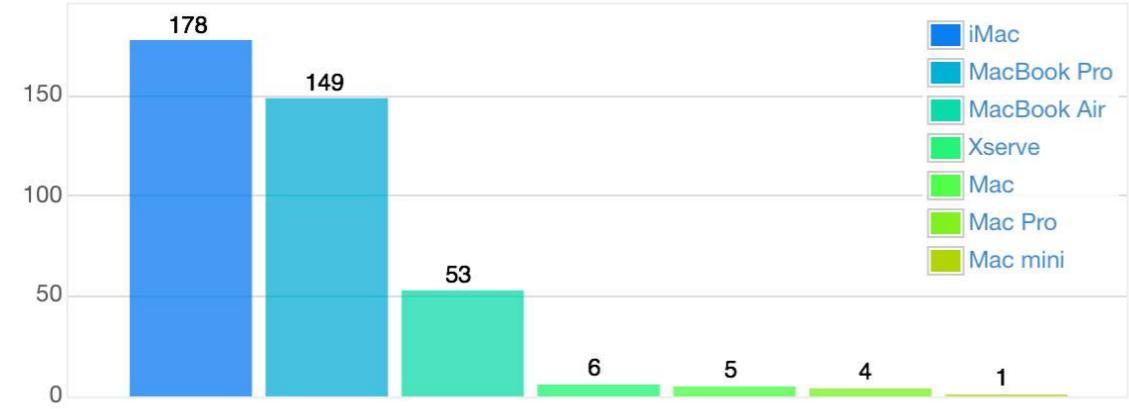
Disk status

311
Under 80%**28**
Over 80%**28**
Over 90%

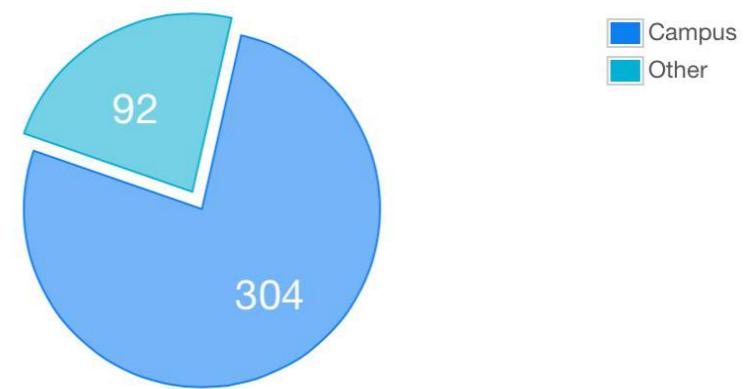
Munki

2
Errors**18**
Warnings**104**
Pending**33**
Installed

Hardware breakdown



Network locations



New clients

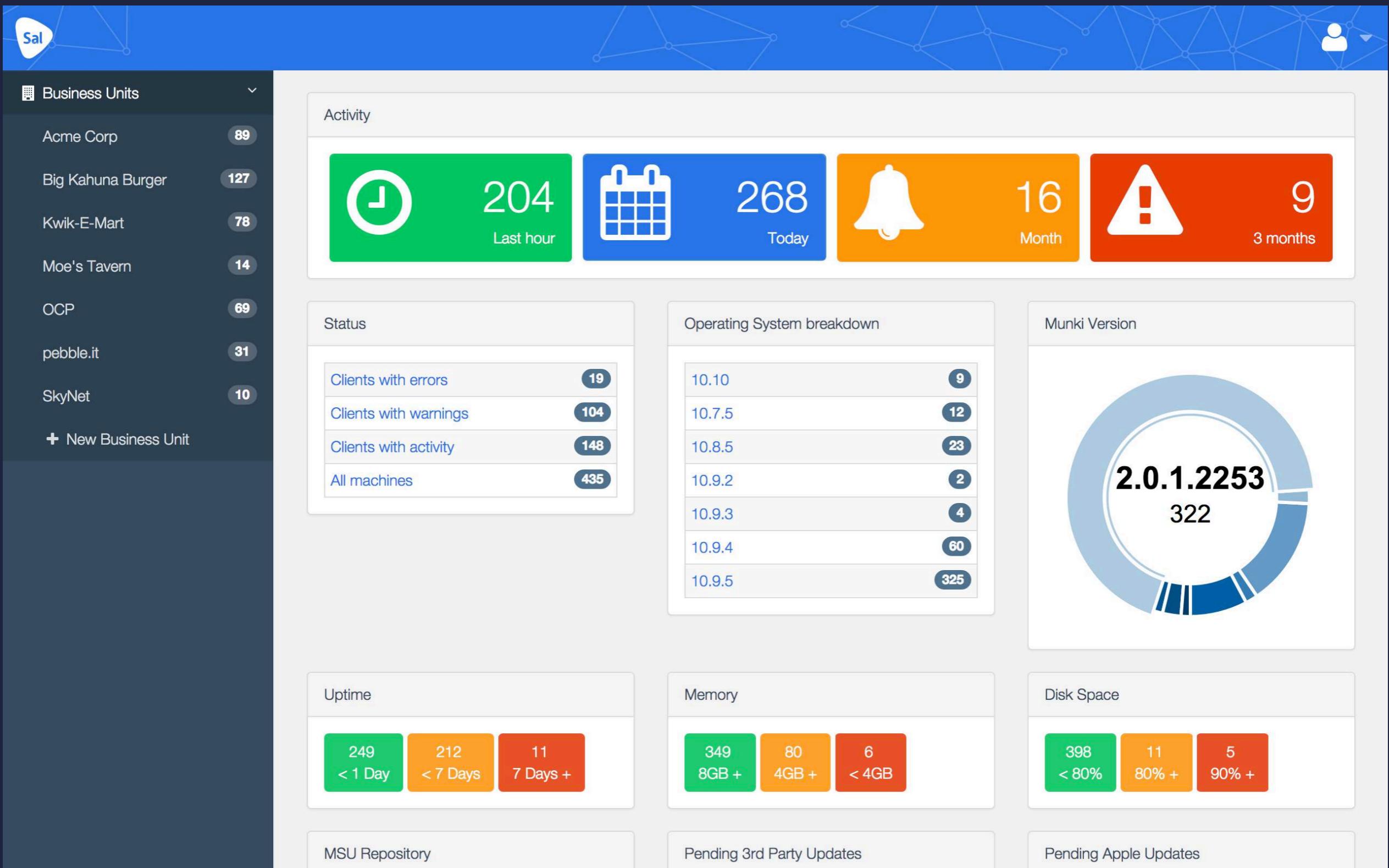
5

Rik's MacBook Pro	a day ago
Jeroen's MacBook Pro	2 days ago
Lorenz Kolbes MacBook Pro	2 days ago
Rob's MacBook Air	6 days ago
mb-016358	6 days ago

Warranty status

Expires in 30 days (Supported)

2



au

MunkiAdmin – Packages

View Open Save Reload Make

REPOSITORY

- All Packages
- Last 30 Days

INSTALLER TYPES

- Copy from Disk Image
- Installer Package
- No Package
- Apple Update Metadata
- Adobe Installer

CATEGORIES

- Uncategorized
- Administration
- Administration
- Apple
- Audio
- Configuration
- Design & CAD
- Developer
- Drivers
- Internet Plugins
- Media Player
- Printers
- Productivity
- Social
- System
- Utilities

Summary

	Pkginfo keys	19
Architectures	0	0
Blocking apps	2	0
Catalogs	2	0
Choices XML	0	0
Installs	1	0
Items to copy	1	0
Receipts	0	0
Requires	0	0
Update for	0	0

Pkginfo

- Location 1Password-4.4.1.plist
- Created 6/5/14, 6:48 AM
- Modified 8/25/14, 9:19 PM
- Last opened 9/4/14, 1:42 AM

Installer

- Location 1Password-4.4.1.dmg
- Created 1/1/70, 1:00 AM
- Modified 6/5/14, 6:48 AM
- Last opened 9/4/14, 9:43 AM

Properties... ?

Name	Display Name	Version	Modified
1Password	1Password	4.4.1	8/25/14
031-0628	OS X Update	1.0	8/25/14
031-0630	OS X Combo Update	1.0	8/25/14
031-3498	OS X Update	10.9.2	8/25/14
031-3499	OS X Update Combined	10.9.2	8/25/14
AdobeAIR	Adobe® Integrated Runtime	13.0.0.111	8/25/14
AdobeAIR	Adobe® Integrated Runtime	14.0.0.110	8/25/14
AdobeAIR	Adobe® Integrated Runtime	14.0.0.178	9/10/14
AdobeAIR	Adobe® Integrated Runtime	15.0.0.249	9/10/14
AdobeFlashPlayer	Adobe Flash Player	13.0.0.214	8/25/14
AdobeFlashPlayer	Adobe Flash Player	14.0.0.125	8/25/14
AdobeFlashPlayer	Adobe Flash Player	14.0.0.145	9/10/14
AdobeFlashPlayer	Adobe Flash Player	14.0.0.176	8/13/14
AdobeFlashPlayer	Adobe Flash Player	15.0.0.152	9/10/14
AdobeReader	Adobe Reader	11.0.07	8/25/14
Alfred2	Alfred 2	2.3	8/25/14
Alfred2	Alfred 2	2.4	9/10/14
AppCleaner	AppCleaner Uninstaller Utility	2.2.3	8/25/14
AppleRemoteDesktop	Apple Remote Desktop	3.0	7/28/14
Atom	Atom	0.113.0	9/10/14
Atom	Atom	0.114.0	9/10/14
Atom	Atom	0.115.0	9/10/14
Atom	Atom	0.117.0	9/10/14
Atom	Atom	0.118.0	9/10/14
Atom	Atom	0.119.0	9/10/14
Atom	Atom	0.120.0	9/10/14
Atom	Atom	0.121.0	9/10/14
Atom	Atom	0.121.0	9/10/14
Atom	Atom	0.122.0	9/10/14
Atom	Atom	0.123.0	9/10/14

Sal

Back

Manage Installs

+ Computer

pebble.it

pebble.it Support 8

pebble.it

Enrollment Key

4jw0h3pq8srek6tfx4uin2iwd1hlv4nvqc8lhz4qh2lcimc581avk49oc3kcz60222rj8uq98mk3ppidwbuiuxnb
yhj3lzvyf6zanoemd5bce4e6a1bf18jor1lyjra

Default Group Edit

pebble.it Support

Managed Installs

Condition_supported_major_os_upgrades	Root
sal_scripts	Root
ADPassMon	pebble.it

Optional Installs

OracleJava8	Root
AutoDMG	pebble.it

Conditional Items

Item	Type	Condition	Source
Hiera	Managed Install	ANY ipv4_address BEGINSWITH '10.30.0'	Root

SUPPORT



Search for topics



+Graham



NEW TOPIC

C

Mark all as read

Filters ▾

munki-dev Shared publicly

30 of many topics ★ 8+1

This is a group for discussing the use of and development of Munki, an OS X software management tool.

Before posting, please check out the documentation here: <https://github.com/munki/munki/wiki> and especially the FAQ here: <https://github.com/munki/munki/blob/master/FAQ.md>

Remember also to search this group for discussions on topics similar to yours. There's a lot of knowledge here.

All are welcome: a new participant's first posting is held for moderation; this prevents unwanted spam.

To join the list, send an email to munki-dev+subscribe@googlegroups.com



Uninstall Adobe CC package existing of several applications fails

By Peter Bos - 25 posts - 721 views



App removal description.

By Greg Batye - 4 posts - 27 views



running munki1 and munki2 with webadmin?

By Ryan Snyder - 2 posts - 30 views



Viewing managed installs in Munki2

By bripanes - 2 posts - 32 views



Get current user in Install Check Script

By Matter - 20 posts - 97 views



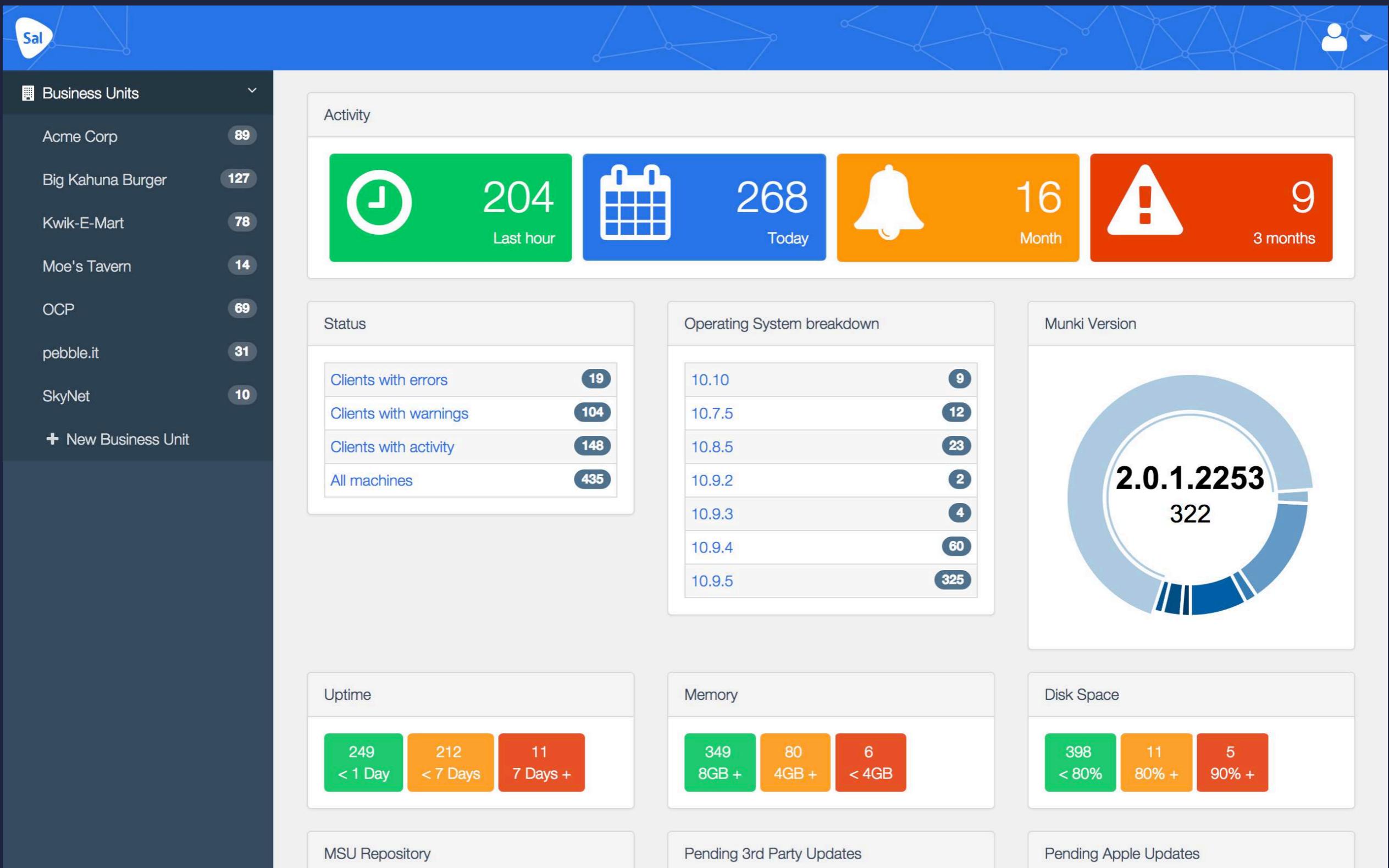
Adobe Acrobat X Pro update

By Jason Hatman - 9 posts - 20 views

pebble.it

All things to all men

<plug>



Sal

Back

Manage Installs

+ Computer

pebble.it

pebble.it Support 8

pebble.it

Enrollment Key

4jw0h3pq8srek6tfx4uin2iwd1hlv4nvqc8lhz4qh2lcimc581avk49oc3kcz60222rj8uq98mk3ppidwbuiuxnb
yhj3lzvyf6zanoemd5bce4e6a1bf18jor1lyjra

Default Group Edit

pebble.it Support

Managed Installs

Condition_supported_major_os_upgrades	Root
sal_scripts	Root
ADPassMon	pebble.it

Optional Installs

OracleJava8	Root
AutoDMG	pebble.it

Conditional Items

Item	Type	Condition	Source
Hiera	Managed Install	ANY ipv4_address BEGINSWITH '10.30.0'	Root

YAY, STICKERS!

pebble.it



**@GRAHAMGILBERT
GRAHAMGILBERT.COM
SALSOFTWARE.COM**