College of Arts and Letters

**CAL Inventory System**

**Software Requirement**

**For Macintosh**

Daehan Chi

Last Update 8/11/2016

**Objectives**

This document is an instruction to demonstrate how the software works on mac system. This system has 4 components, 2 bash script, 1 application, and 1 plist.

**Codes**

**Filename: inventory\_update.plist**

**Location: /Library/LaunchAgent/**

**Permission: 600 root**:**wheel** **\*\*IMPORTANT\*\***

1 <?xml version="1.0" encoding="UTF-8"?>

2 <!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">

3 <plist version="1.0">

4 <dict>

5 <key>Disabled</key>

6 <false/>

7 <key>Label</key>

8 <string>edu.csusb.update\_inventory</string>

9 <key>RunAtLoad</key>

10 <true/>

11 <key>ProgramArguments</key>

12 <array>

13 <string>/usr/bin/open</string>

14 <string>-W</string>

15 <string>/Users/Shared/inventory\_update.app </string>

16 </array>

17 </dict>

18 </plist>

**Description**:

This XML code sets environment for launchd.

This .plist file sets (1) the label of the process - “edu.csusb.update\_inventory”

(2) What program you want to execute - /Users/Shared/inventory\_update.app

(3) When to run the program – every time any user login.

Since this .plist file is in /Library/LaunchAgent/, this setting affect to all usesrs. If this file is in ~/Library/LaunchAgent/, it will affect to the particular user.

**Filename: inventory\_update.sh**

1 echo $('date') >> /Users/Shared/$('whoami')\_log\_inv.txt

2 curl --request POST 'https://cal.csusb.edu/ip/include/update\_inventory.php' --data "hostname=$(hostname -s)&ip=$(ifconfig en0 | awk '$1 == "inet" {print $2}')&mac=$(ifconfig en0 | awk '$1 == "ether" {print $2}')&AvailableMemory=$(top -l 1 -s 0 | grep PhysMem | awk $1=="PhysMem:"{print $6}')&HD-Percent=(df -h | awk 'NR==2 {print $5}')" >> /Users/Shared/$('whoami')\_log\_inv.txt

**Description**:

First command makes an individual account preventing permission encounters of different users by making new text file belong to each account.

Second command sends a request from the mac to the inventory system.

This script goes to “Automator” application, with “Run a script” library, and save as a app, so this script can be ran as app by “inventory\_update.plist”

\*Do not copy and paste including the numbers indicating the n-th line and the line-breaking spaces.

**Filename: inventory\_setup.sh**

#!/bin/bash

#<----------MANUAL INPUT--------🡪

#TAKE TAG NUMBER

2 echo "Input tag number(5): "

3 read TAG

#TAKE BUILDING

4 echo "Input building(2): "

5 read BUILDING

#TAKE ROOM NUMBER

6 echo "Input room number: "

7 read ROOM

#TAKE PURPOSE

8 echo "Input purpose(Faculty, Name, Lab, etc): "

9 read PURPOSE

10 echo -e "Tag number:\t" $TAG

11 echo -e "Location:\t" $BUILDING$ROOM

12 echo -e "Purpose:\t" $PURPOSE

#<----------AUTOMATED INPUT--------🡪

#1. TAKE HOSTNAME

13 HOSTNAME=$(hostname -s)

14 echo "1. PRINT HOSTNAME: $HOSTNAME"

#2. TAKE IP ADDRESS

15 IP=$(ifconfig en0 | awk '$1 == "inet" {print $2}')

16 echo "2. PRINT IP ADDRESS: $IP"

#3. TAKE MAC ADDRESS

17 MAC=$(ifconfig en0 | awk '$1 == "ether" {print $2}')

18 echo "3. PRINT MAC ADDRESS: $MAC"

#4. SET ACTIVE TO 1

19 ACTIVE="1"

20 echo "4. PRINT ACTIVE: $ACTIVE"

#5. TAKE OPERATING SYSTEM

21 OS=$(system\_profiler SPSoftwareDataType | awk '$1 == "System" && $2 == "Version:" {print $3" "$4" "$5" "$6}')

22 echo "5. PRINT OS VERISON: $OS"

#6. TAKE TOTAL MEMORY

23 TOTALMEM=$(system\_profiler SPHardwareDataType | awk '$1 == "Memory:" {print $2" "$3}')

24 echo "6. PRINT TOTAL MEMORY: $TOTALMEM"

#7. TAKE AVAILABLE MEMORY

25 AVAILABLEMEM=$(top -l 1 -s 0 | awk '/PhysMem/ || $1=="PhysMem:"{print $10}')

26 echo "7. PRINT AVAILABLE MEMORY: $AVAILABLEMEM"

#8. TAKE SYSTEM MODEL

SYSMO27 DEL=$(system\_profiler SPHardwareDataType | awk '$1=="Model" && $2=="Identifier:" {print $3}')

29 echo "8. PRINT SYSTEM MODEL: $SYSMODEL"

#9. TAKE TOTAL HARD DRIVE SIZE

30 TOTALHD=$(df -h | awk 'NR==2 {print $2}')

31 echo "9. PRINT TOTAL HARD DRIVE: $TOTALHD"

#10. TAKE AVAILABLE HARD DRIVE PERCENT

32 PERCENTHD=$(df -h | awk 'NR==2 {print $5}')

33 echo "10. PRINT PERCENT OF AVAILEABLE HARD DRIVE: $PERCENTHD"

#11. TAKE SERIAL NUMBER

34 SERIAL=$(system\_profiler SPHardwareDataType | awk '$1=="Serial" {print $4}')

35 echo "11. PRINT SERIAL NUMBER: $SERIAL"

36 curl --request POST 'https://cal.csusb.edu/ip/include/add\_inventory.php' --data "tag=$TAG&building=$BUILDING&room=$ROOM&purpose=$PURPOSE&hostname=$HOSTNAME&ip=$IP&mac=$MAC&active=$ACTIVE&os=$OS&TotalMemory=$TOTALMEM&AvailableMemory=$AVAILABLEMEM&SysModel=$SYSMODEL&HD-Total=$TOTALHD&HD-Percent=$PERCENTHD&serial=$SERIAL"

**Description**:

This script initialize the computer on the inventory database by collecting manual and automated inputs and sending them to the PHP server.

**Filename: run\_this\_script.sh**

1 bash ~/Desktop/inventory\_setup.sh

2 sudo cp ~/Desktop/inventory\_update.plist /Library/LaunchAgents/

3 sudo chown root:wheel /Library/LaunchAgents/inventory\_update.plist

4 sudo chmod 600 /Library/LaunchAgents/inventory\_update.plist

5 sudo mv ~/Desktop/inventory\_update.app /Users/Shared/

6 sudo chmod 755 /Users/Shared/inventory\_update.app

7 launchctl load /Library/LaunchAgents/inventory\_update.plist

8 launchctl list | grep csusb

**Description**:

This script is the only script that should be run by hands.

(1) It runs “inventory\_setup.sh” and setup initialized status on inventory.

(2) move inventory\_update.plist to /Library/LaunchAgents/ and change ownership and permission to root:wheel and 600. (\*the file should be on the Desktop before\*)

(3) move inventory\_update.app to /Users/Shared/ and change the permission to 755. (\*the file should be on the Desktop before\*)

(4) load inventory\_update.plist by launchctl and find the process on the list.

**Procedure**

1. Copy 4 items in “/dist” folder to ~/Desktop. (inventory\_setup.sh, inventory\_update.app, inventory\_update.plist, run\_this\_script.sh)
2. Open a terminal and $bash ~/run\_this\_script.sh

(\*Do not SUDO\*) – this will sudo launchctl to work for Daemon not Agent.