



POS Monitor Upgrade Solution for Linux User's Guide

Version: 1.0

May 2013

This edition applies to Version 1.0 of the POS Monitor Upgrade Solution for Linux and to all subsequent releases and modifications until otherwise indicated in new editions.

Toshiba Global Commerce Solutions, Inc. documentation is available at: www.toshibagcs.com/support. Under popular links, select **Publications**.

Address your comments to:

TOSHIBA GLOBAL COMMERCE SOLUTIONS, INC.
A Toshiba and IBM Company
PO BOX 12195, BLDG 307/DEPT. VC2A 3039 CORNWALLIS RD,
RESEARCH TRIANGLE PARK, NC 27709-2195 USA

When you send information to TOSHIBA, you grant TOSHIBA a nonexclusive right to use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

2013 © Copyright Toshiba global Commerce Solutions, Inc.

Table of Contents

Table of Contents	2
Version History	3
1 Introduction to MUS	4
1.1 What is Monitor Upgrade Solution	4
1.2 MUS Design Goals	4
1.2.1 Ease of Use	4
1.2.2 Less user Intervention	4
1.2.3 Local System Update Solution.....	4
1.2.4 Safe Firmware Upgrade	4
2 MUS Overview	5
2.1 Pre Requisites	5
2.1.1 Linux OS Supports.....	5
2.1.1 Python 2.6 Environment	5
2.1 .2 ELO Touch Flash Utility	5
2.2 Operational Package (OP).....	5
2.2.1 File format	5
2.2. 2 Components of OP.....	5
2.3 Firmware Upgrade Package (UP)	6
2.3.1 File format	6
2.3.2 Components of UP.....	6
3 Using MUS	7
3.1 OP Installation	7
3.2 Checking if Monitor Upgrade Service Init script is installed.....	8
3.3 UP Installation	8
3.4 Checking if the upgrade configuration files are extracted	9
3.5 Getting the Logs.....	10
3.6 MUS Uninstall	12
3.7 Extracting only the files inside an OP or UP package.....	13
4 MUS Directory Structure	14
4.1 MUS base directory	14
4.2 Configurations directory	15
4.3 Logs Directory	15
5 Frequently Ask Questions (FAQ).....	16
6 Known Limitations	17

Version History

<i>Version</i>	<i>Date</i>	<i>Change Description</i>
<i>1.0</i>	1 May 2013	Initial Release

1 Introduction to MUS

1.1 What is Monitor Upgrade Solution

The Monitor Upgrade Solution or MUS for short is the new solution by Toshiba Global Commerce Solutions, Inc. to perform *silent and unattended* updating of the Touch and Video Firmware of Toshiba Monitor types 4820_21x51x and 4820_2Lx5Lx .

1.2 MUS Design Goals

This solution is made in order to achieve a set of design goals which we think will give more value add to our customer satisfaction.

1.2.1 Ease of Use

Because the solution is a self-extracting archive file, the user does not need to set any environment variables for this to work. It will have its own script to do those things. The user will need to have a root access and execute the package of the solution. After that, the Monitor Upgrade service is already installed in the service. It will activate every time there is a system reboot. However it will only do firmware updates when it sees there is an available upgrade package in the system.

1.2.2 Less user Intervention

Since the script runs on boot-up, there is no user intervention required to do the firmware upgrading. The user will only notice that the firmware update has taken place when the monitor blacks out and that it is currently updating. The log files in the MUS specific directory can be checked to see if the update was successful or if it failed. If it failed, the upgrade package will need to be run again to trigger the script to do the update once the system reboots.

1.2.3 Local System Update Solution

As compare to the remote update solution, the MUS is activated in the target system on every reboot. This means that the user will have to reboot the system locally to trigger an update. With the remote update solution, the update is triggered by a remote user.

1.2.4 Safe Firmware Upgrade

Since the firmware update is done only during the system reboot and not anytime, this will avoid having an update during the middle of the store hours or store operations. With this it can safely be done when the machine reboots before the store opens or closes.

2 MUS Overview

Monitor Upgrade Solution is composed of two installation packages. First is the Operational Package (OP), this is the one needed to be installed to the system for the whole MUS to operate. The second is the Upgrade Package (UP), this only needs to be installed this when there is a new firmware available for upgrading.

2.1 Pre Requisites

The following are the things that Monitor Upgrade Solution needs to work:

1. Supported Linux Operating System
2. Python 2.6
3. Elo Kernel Mode Package

2.1.1 Linux OS Supports

- SLES 11 SP2 32-Bit
- SLED 11 SP2 32-Bit
- SLEPOS 11 SP2 32-Bit

2.1.1 Python 2.6 Environment

Python 2.6 is a default package included with SLES 11 and SLED 11. On SLEPOS sometimes it might not have been included Python 2.6 environment with it. If it is not installed, then it will need to be installed since the Monitor Upgrade Scripts are running under python.

The latest version of Python 2.6 can be downloaded from the following link:

<http://www.python.org/download/releases/2.6.8/>

2.1.2 ELO Touch Flash Utility

Elo is the vendor that supplies the needed Touch firmware update drivers for the monitor. Elo also provides the installation utility to install these drivers. They also provide the Touch Flash application included in the MUS Operational Package.

In order to update the firmware, one of the following two kernel mode driver packages will need to download from Elo website:

http://www.elotouch.com/Support/Downloads/dnld_company.asp

Download the package which matches the kernel configuration on your machine.

1. **elofousb-kmp-default-1.0_3.0.13_0.27-4.i586.rpm** - For “default” configurations
2. **elofousb-kmp-pae-1.0_3.0.13_0.27-4.i586.rpm** - For “pae” configurations

2.2 Operational Package (OP)

The Operational Package will install all the components needed for daily use. This daily use functions to be supported are:

1. Enabling the operator to use the touch screen.
2. Enabling the updating of the touch configuration at boot.
3. Providing inventory information to remote machines via RMA.
4. Performing firmware updates at boot time

2.2.1 File format

The Operational Package is a self-extracting archive file with the extension (.bz2.run). The ".run" signifies that it is an executable file and needs to be run by the user. The user must have “root” access to be able to install the package.

Example: **4820_Operational_Package-1.0.00.bz2.run**

2.2.2 Components of OP

The components included in the Operational Package are:

1. Video firmware update utility - **Sampo 4820MonUtil**
2. Touch firmware update utility scripts - **EloConfig and EloDownload**
3. Toshiba Monitor Inventory RPM
4. Toshiba Monitor Upgrade Service RPM

2.3 Firmware Upgrade Package (UP)

The Firmware Upgrade Package will provide the updated touch and video firmware files along with needed configuration files to enable the firmware updates at boot time. The upgrade will install any update components copied onto the machine by the Operational Package.

2.3.1 File format

The Upgrade Package is also a self-extracting archive file with the extension (.bz2.run). The ".run" signifies that it is an executable file and needed to be run by the user. The user must have “root” access to be able to do so.

Example: ***4820_Upgrade_Package-1.0.00.bz2.run***

2.3.2 Components of UP

Depending upon the firmware upgrade package type, the following should be included:

1. Video firmware update file
2. Video firmware update scripts and configuration file
3. Touch firmware update file
4. Touch firmware update scripts and configuration file

If an update to the firmware update driver/component provided by the Operational Package is needed, the Firmware Upgrade Package will include the required driver/component.

3 Using MUS

3.1 OP Installation

Once the OP package has been downloaded, the package can be installed in a terminal window as the "root" user. Listing 1 shows, how to run the OP installation.

```
fliwa@suse: /home/fliwa/Desktop/OP # ls -al
total 796
drwxr-xr-x  2 root  root   4096 Dec  7 18:08 .
drwxr-xr-x 15 fliwa users 4096 Dec  7 18:08 ..
-rwxr-xr-x  1 root  root 800068 Dec  5 18:03 4820_Operational_Package-1.0.00.bz2.run
fliwa@suse: /home/fliwa/Desktop/OP # ./4820_Operational_Package-1.0.00.bz2.run
Verifying archive integrity... All good.
Uncompressing 4820 TOSHIBA MONITOR OPERATIONAL PACKAGE....
#####
#                               Installing OP RPMs now!                               #
#####
Installing...  toshiba-dspdrv-suse-1.2.1-1.0.i586.rpm
Preparing...   #####

SFCB Cim Server found
toshiba-dspdrv-suse   #####
SFCB Cim Server found
SFCB Cim Server already stopped

#####
#                               Package is installed Succesfully!                               #
#####

Installing...  toshiba-monitor_upgrade-1.0.0-1.0.i586.rpm
Preparing...   #####

toshiba-monitor_upgrade   #####

Starting Monitor Upgrade Service installation...

Checking dependencies...
Extracting  ELO TOOLS...
Extracting  SAMPO TOOLS...
Creating MUS directory...
Creating scripts directories...
Creating soft-link to scripts...
Linking init scripts...
Linking monitorupgrade...
Registering MonitorUpgradeSvc...

Installation Complete!

#####
#                               Package is installed Succesfully!                               #
#####
```

Listing 1

3.2 Checking if Monitor Upgrade Service Init script is installed

This is to verify after installing the OP, if the MonitorUpgradeSvc is actually installed in the system. Listing 2 shows the MonitorUpgradeSvc is installed but is not currently running. This is correct, because it will only run when the systems reboots. After performing its task it will stop.

```
fliwa-suse:/home/fliwa/Desktop/OP # chkconfig -l MonitorUpgradeSvc
MonitorUpgradeSvc      0:off  1:off  2:off  3:on   4:off  5:on   6:off
fliwa-suse:/home/fliwa/Desktop/OP # /etc/init.d/MonitorUpgradeSvc status
Program isn't running
```

Listing 2

3.3 UP Installation

When the user received a new firmware update, it will come inside an Upgrade Package. Listing 3 shows how to install an Upgrade package.

```
fliwa-suse:/home/fliwa/Desktop/UP # ls -l
total 64
-rwxr-xr-x 1 root root 59938 Dec  7 18:28 4820_Upgrade_Package-1.0.00.bz2.run
fliwa-suse:/home/fliwa/Desktop/UP # ./4820_Upgrade_Package-1.0.00.bz2.run
Verifying archive integrity... All good.
Uncompressing 4820 TOSHIBA MONITOR UPGRADE PACKAGE.....
Creating staging directory...
making /opt/toshiba/MonitorUpgrade/up
making /opt/toshiba/MonitorUpgrade/up/4820
making /opt/toshiba/MonitorUpgrade/up/4820/stage
staging files now...
running preupdate now...

Upgrade Package Successfully loaded!

#####
#           Please reboot the system for the update to take effect.           #
#####
fliwa-suse:/home/fliwa/Desktop/UP #
```

Listing 3

3.4 Checking if the upgrade configuration files are extracted

After installing the UP, a user can check if it was successfully extracted in to the system. Listing 4, shows what is inside the configuration file. The configuration file can be found in the `/etc/opt/toshiba/MonitorUpgrade/CONFIGS` directory.

```
Fliva-suse:/etc/opt/toshiba/MonitorUpgrade/CONFIGS/mus/update_task # ls -al
total 12
drwxr-xr-x 2 root root 4096 Dec  7 18:35 .
drwxr-xr-x 3 root root 4096 Dec  7 18:10 ..
-rw-r--r-- 1 root root  349 Dec  3 11:55 updatetask.conf
Fliva-suse:/etc/opt/toshiba/MonitorUpgrade/CONFIGS/mus/update_task # cat updatetask.conf
#####
#Everything MUS needs to know about a UP task #
#####

[Identity]
#
#
#
SUPName=4820_21X_51X_TOUCH_DRIVER_UPDATE_VERSION_1.0.0__0
Uuid={78E2A472-14D7-44CE-B3E5-E6C56BCF3324}

[Task]
#
#
TaskName=EloAnakFlashTask

[Firmware]
Path='FIRMWARE_0_N.TXT'

Fliva-suse:/etc/opt/toshiba/MonitorUpgrade/CONFIGS/mus/update_task #
```

Listing 4

3.5 Getting the Logs

The logs for the MUS is found under the following directory: `/var/log/toshiba/MonitorUpgrade/LOGS`

Listing 5.1 shows the log for the OP Installation.

```
tlwa@suse: /var/log/toshiba/MonitorUpgrade/LOGS/op_installation # cat OP_INSTALL-2012-12-07_18:10:49.652941.log
2012-12-07 18:10:49,653 - [OP_INSTALL_LOG]:[INFO] - Starting installation...
2012-12-07 18:10:49,653 - [OP_INSTALL_LOG]:[INFO] - Checking dependencies...
2012-12-07 18:10:49,653 - [OP_INSTALL_LOG]:[INFO] - lsmod | grep elo
2012-12-07 18:10:49,669 - [OP_INSTALL_LOG]:[INFO] - elousb                12642  0
2012-12-07 18:10:49,669 - [OP_INSTALL_LOG]:[INFO] - usbcore                185642  6 usb_storage,uas,usbhid,elousb,ehci_hcd
2012-12-07 18:10:49,669 - [OP_INSTALL_LOG]:[INFO] - Extracting ELO TOOLS...
2012-12-07 18:10:49,670 - [OP_INSTALL_LOG]:[INFO] - tar -C /opt/toshiba/MonitorUpgrade/op/4820/ELO/ -xvf /opt/toshiba/MonitorUpgrade/op/4820/ELO/elo.tar
2012-12-07 18:10:49,689 - [OP_INSTALL_LOG]:[INFO] - EcBatch.sh
2012-12-07 18:10:49,689 - [OP_INSTALL_LOG]:[INFO] - eloconfig
2012-12-07 18:10:49,689 - [OP_INSTALL_LOG]:[INFO] - EloConfig.sh
2012-12-07 18:10:49,689 - [OP_INSTALL_LOG]:[INFO] - elodownload
2012-12-07 18:10:49,690 - [OP_INSTALL_LOG]:[INFO] - EloDownload.sh
2012-12-07 18:10:49,690 - [OP_INSTALL_LOG]:[INFO] - EloEcho.sh
2012-12-07 18:10:49,690 - [OP_INSTALL_LOG]:[INFO] - FwTouchUpgradeUtil
2012-12-07 18:10:49,694 - [OP_INSTALL_LOG]:[INFO] - getfw.sh
2012-12-07 18:10:49,694 - [OP_INSTALL_LOG]:[INFO] - SUSE -- READ ME NOW.doc
2012-12-07 18:10:49,694 - [OP_INSTALL_LOG]:[INFO] - SUSE -- READ ME NOW.odt
2012-12-07 18:10:49,694 - [OP_INSTALL_LOG]:[INFO] - udev_namer
2012-12-07 18:10:49,695 - [OP_INSTALL_LOG]:[INFO] - Extracting SAMPO TOOLS...
2012-12-07 18:10:49,695 - [OP_INSTALL_LOG]:[INFO] - tar -C /opt/toshiba/MonitorUpgrade/op/4820/SAMPO/ -xvf /opt/toshiba/MonitorUpgrade/op/4820/SAMPO/sampo.tar
2012-12-07 18:10:49,702 - [OP_INSTALL_LOG]:[INFO] - 4820MonUtil
2012-12-07 18:10:49,703 - [OP_INSTALL_LOG]:[INFO] - rc.local
2012-12-07 18:10:49,703 - [OP_INSTALL_LOG]:[INFO] - readme.txt
2012-12-07 18:10:49,703 - [OP_INSTALL_LOG]:[INFO] - Release note V1.12.txt
2012-12-07 18:10:49,703 - [OP_INSTALL_LOG]:[INFO] - Creating MUS directory...
2012-12-07 18:10:49,703 - [OP_INSTALL_LOG]:[INFO] - Checking directory -/opt/toshiba/MonitorUpgrade/mus/scripts/...
2012-12-07 18:10:49,704 - [OP_INSTALL_LOG]:[INFO] - /opt/toshiba/MonitorUpgrade/mus/scripts/ exist!
2012-12-07 18:10:49,704 - [OP_INSTALL_LOG]:[INFO] - Creating scripts directories
2012-12-07 18:10:49,704 - [OP_INSTALL_LOG]:[INFO] - mkdir /opt/toshiba/MonitorUpgrade/mus/scripts/init
2012-12-07 18:10:49,707 - [OP_INSTALL_LOG]:[INFO] - mkdir /opt/toshiba/MonitorUpgrade/mus/scripts/monitorupgrade
2012-12-07 18:10:49,710 - [OP_INSTALL_LOG]:[INFO] - Creating soft-link to scripts...
2012-12-07 18:10:49,710 - [OP_INSTALL_LOG]:[INFO] - ln -sf /usr/local/lib/python2.6/site-packages/init/* /opt/toshiba/MonitorUpgrade/mus/scripts/init/
2012-12-07 18:10:49,716 - [OP_INSTALL_LOG]:[INFO] - ln -sf /usr/local/lib/python2.6/site-packages/monitorupgrade/* /opt/toshiba/MonitorUpgrade/mus/scripts/monitorupgrade/
2012-12-07 18:10:49,719 - [OP_INSTALL_LOG]:[INFO] - Adding init script to /etc/init.d/...
2012-12-07 18:10:49,719 - [OP_INSTALL_LOG]:[INFO] - cp -p /usr/local/lib/python2.6/site-packages/initscript/MonitorUpgradeSvc.py /etc/init.d/MonitorUpgradeSvc
2012-12-07 18:10:49,723 - [OP_INSTALL_LOG]:[INFO] - chmod +x /etc/init.d/MonitorUpgradeSvc
2012-12-07 18:10:49,726 - [OP_INSTALL_LOG]:[INFO] - Registering MonitorUpgradeSvc...
2012-12-07 18:10:49,726 - [OP_INSTALL_LOG]:[INFO] - chkconfig -a MonitorUpgradeSvc
2012-12-07 18:10:49,831 - [OP_INSTALL_LOG]:[INFO] - MonitorUpgradeSvc          0:off 1:off 2:off 3:on  4:off 5:on  6:off
2012-12-07 18:10:49,833 - [OP_INSTALL_LOG]:[INFO] - chkconfig MonitorUpgradeSvc on
2012-12-07 18:10:49,883 - [OP_INSTALL_LOG]:[INFO] - Installation Complete!
```

Listing 5.1

Listing 5.2 shows an example log for the MonitorUpgradeSvc running.

```
fliva-suse: /var/log/toshiba/MonitorUpgrade/LOGS/mus # ls -l
total 8
-rw-r--r-- 1 root root 6704 Dec 7 18:55 MonitorUpgrade-2012-12-07 18:54:39.006998.log
fliva-suse: /var/log/toshiba/MonitorUpgrade/LOGS/mus # cat MonitorUpgrade-2012-12-07 18:54:39.006998.log
2012-12-07 18:54:39,007 - [MUS_LOG-v1.0.0]:[INFO] - T O S H I B A   G C S   M O N I T O R   U P G R A D E   S O L U T I O N

2012-12-07 18:54:39,007 - [MUS_LOG-v1.0.0]:[INFO] - ===== START OF LOG =====

2012-12-07 18:54:39,007 - [MUS_LOG-v1.0.0]:[INFO] - MonitorUpgradeSvc:start(): Starting Toshiba GCS Monitor Upgrade Service...
2012-12-07 18:54:39,008 - [MUS_LOG-v1.0.0]:[INFO] - MonitorUpgradeSvc:lock(): Creating /var/lock/subsys/TSKDRVSVC_{78330F4D-8A76-45BB-BF1F-A33843EB217D} lock file
2012-12-07 18:54:39,008 - [MUS_LOG-v1.0.0]:[INFO] - App Constructed!
2012-12-07 18:54:39,008 - [MUS_LOG-v1.0.0]:[INFO] - App::run()-START
2012-12-07 18:54:39,008 - [MUS_LOG-v1.0.0]:[INFO] - App::run()-Creating TaskManager Object
2012-12-07 18:54:39,008 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr Constructed!
2012-12-07 18:54:39,008 - [MUS_LOG-v1.0.0]:[INFO] - App::run()-Calling TaskMgr::initialize()
2012-12-07 18:54:39,008 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::initialize-Initialized OK
2012-12-07 18:54:39,008 - [MUS_LOG-v1.0.0]:[INFO] - App::run()-Calling TaskMgr::performTasks()
2012-12-07 18:54:39,008 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::performTasks()===== START

2012-12-07 18:54:39,008 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::isInRecoveryMode-----START-----
2012-12-07 18:54:39,009 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::isInRecoveryMode-Checking if system is in recovery mode...
2012-12-07 18:54:39,009 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::checkRecoveryFile-----START-----
2012-12-07 18:54:39,009 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::checkRecoveryFile-Checking Sampo Recovery trigger file...
2012-12-07 18:54:39,009 - [MUS_LOG-v1.0.0]:[INFO] - doesFileExist()-Found! /etc/init.d/after.local
2012-12-07 18:54:39,009 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::checkRecoveryFile-Found Sampo Recovery trigger file...
2012-12-07 18:54:39,009 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::checkRecoveryFile-----END-----

2012-12-07 18:54:39,025 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::isInRecoveryMode()- Not in Recovery Mode - Sampo 4820MonUtil recovery string is not in file
2012-12-07 18:54:39,025 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::isInRecoveryMode-----END-----

2012-12-07 18:54:39,025 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::checkUpdateTaskFile-----START-----
2012-12-07 18:54:39,025 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::checkUpdateTaskFile-Checking Update task file...
2012-12-07 18:54:39,221 - [MUS_LOG-v1.0.0]:[INFO] - doesFileExist()-Found! /etc/opt/toshiba/MonitorUpgrade/CONFIGS/mus/update_task/updatetask.conf
2012-12-07 18:54:39,222 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::checkUpdateTaskFile-Found Update task file...
2012-12-07 18:54:39,222 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::checkUpdateTaskFile-----END-----

2012-12-07 18:54:39,222 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::performTasks()- Found an Update task available-checking task file now...
2012-12-07 18:54:39,222 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::performTasks() -Calling MUSConfigParser()
2012-12-07 18:54:39,222 - [MUS_LOG-v1.0.0]:[INFO] - UpdateTaskConfig Constructed
2012-12-07 18:54:39,222 - [MUS_LOG-v1.0.0]:[INFO] - UpdateTaskConfig -Calling MUSConfigParser()
2012-12-07 18:54:39,222 - [MUS_LOG-v1.0.0]:[INFO] - MUSConfigParser Constructed
2012-12-07 18:54:39,222 - [MUS_LOG-v1.0.0]:[INFO] - UpdateTaskConfig -Reading configuration file /etc/opt/toshiba/MonitorUpgrade/CONFIGS/mus/update_task/updatetask.conf
2012-12-07 18:54:39,248 - [MUS_LOG-v1.0.0]:[INFO] - UpdateTaskConfig::getTaskName() - task_name = EloAnakFlashTask
2012-12-07 18:54:39,248 - [MUS_LOG-v1.0.0]:[INFO] - UpdateTaskConfig::getFirmwarePath() - fw_path = 'FIRMWARE_0_N.TXT'
2012-12-07 18:54:39,248 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::performTasks()- TaskName= EloAnakFlashTask
2012-12-07 18:54:39,248 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::performTasks()- Calling doEloAnakFlashTask()
2012-12-07 18:54:39,248 - [MUS_LOG-v1.0.0]:[INFO] - EloAnakFlashTask Constructed
2012-12-07 18:54:39,248 - [MUS_LOG-v1.0.0]:[INFO] - EloAnakFlashTask::run() -----START-----
2012-12-07 18:54:39,268 - [MUS_LOG-v1.0.0]:[INFO] - doesFileExist()-Found! /opt/toshiba/MonitorUpgrade/op/4820/ELO/EloDownload.sh
2012-12-07 18:54:59,786 - [MUS_LOG-v1.0.0]:[INFO] - Subprocess[ELODOWNLOAD-ANAK] OUTPUT--><root>

2012-12-07 18:55:01,056 - [MUS_LOG-v1.0.0]:[INFO] - Subprocess[ELODOWNLOAD-ANAK] OUTPUT--> <WORKER NAME="EloDownload" VERSION="2.08">
2012-12-07 18:55:01,056 - [MUS_LOG-v1.0.0]:[INFO] - Subprocess[ELODOWNLOAD-ANAK] OUTPUT--> <RESPONSE RESULT="SUCCESS" CONTROLLER="A28540">
2012-12-07 18:55:01,056 - [MUS_LOG-v1.0.0]:[INFO] - Subprocess[ELODOWNLOAD-ANAK] OUTPUT--> </RESPONSE>
2012-12-07 18:55:01,056 - [MUS_LOG-v1.0.0]:[INFO] - Subprocess[ELODOWNLOAD-ANAK] OUTPUT--> </WORKER>
2012-12-07 18:55:01,056 - [MUS_LOG-v1.0.0]:[INFO] - Subprocess[ELODOWNLOAD-ANAK] OUTPUT--></root>

2012-12-07 18:55:01,056 - [MUS_LOG-v1.0.0]:[INFO] - EloAnakFlashTask::run() -----END-----

2012-12-07 18:55:01,056 - [MUS_LOG-v1.0.0]:[INFO] - TaskMgr::performTasks()===== END

2012-12-07 18:55:01,056 - [MUS_LOG-v1.0.0]:[INFO] - cleanUp()-----START-----
2012-12-07 18:55:01,057 - [MUS_LOG-v1.0.0]:[INFO] - doesFileExist()-Found! /etc/opt/toshiba/MonitorUpgrade/CONFIGS/mus/update_task/updatetask.conf
2012-12-07 18:55:01,057 - [MUS_LOG-v1.0.0]:[INFO] - rm /etc/opt/toshiba/MonitorUpgrade/CONFIGS/mus/update_task/updatetask.conf
2012-12-07 18:55:01,057 - [MUS_LOG-v1.0.0]:[INFO] - rm -rf /opt/toshiba/MonitorUpgrade/mus/update_task/fw_temp/*.*
2012-12-07 18:55:01,126 - [MUS_LOG-v1.0.0]:[INFO] - cleanUp()-----END-----
```

Listing 5.2

3.6 MUS Uninstall

To uninstall the whole Monitor Upgrade Solution run the **MUS_UnInstall** script in a terminal window as “root”. The Uninstallation script is located at **/opt/toshiba/UnInstall** folder. Listing 6 shows the output of running the script.

```
flwa-suse:/opt/toshiba/UNINSTALL # ls -l
total 8
-rwxr-xr-x 1 root root 5753 Dec  5 14:55 MUS_UnInstall
flwa-suse:/opt/toshiba/UNINSTALL # ./MUS_UnInstall
UnInstalling Toshiba 4820 OPERATIONAL PACKAGE now...
rpm -qa | grep toshiba-dspdrv-suse >> /usr/tmp/toshiba/rpmlist.txt

rpm -qa | grep toshiba-monitor_upgrade >> /usr/tmp/toshiba/rpmlist.txt

UnInstalling... toshiba-dspdrv-suse-1.2.1-1.0
rpm -ev toshiba-dspdrv-suse-1.2.1-1.0
SFCB Cim Server found

SFCB Cim Server already stopped

#####
#                               UnInstallation Complete!                               #
#####

UnInstalling... toshiba-monitor_upgrade-1.0.0-1.0
rpm -ev toshiba-monitor_upgrade-1.0.0-1.0
Uninstalling Toshiba Monitor Upgrade...

Deleting Monitor Upgrade folder...

Removing MonitorUpgradeSvc

MonitorUpgradeSvc      0:off 1:off 2:off 3:off 4:off 5:off 6:off

Uninstall Complete!

#####
#                               UnInstallation Complete!                               #
#####
```

Listing 6

3.7 Extracting only the files inside an OP or UP package

For OS like SLEPOS, users only need the files and RPM packages inside the OP or UP in order for them to be able to package those files themselves inside the stripped-down SLEPOS Image. Therefore, a method is needed to extract only those needed files. Listing 7 shows how to extract the files only.

Syntax :

```
OP : ./4820_Operational_Package-x.x.xx.bz2.run --noexec --target <target directory>
UP : ./4820_21x_51x_Touch_FW_Update-x.x.xx.bz2.run --noexec --target <target directory>
```

Note:

*--noexec - means you don't execute the scripts inside the package
<target directory> - this is where you want to extract the files.*

```
fliwa-suse:/home/fliwa/Desktop/Latest_OP_UP # ./4820_Operational_Package-1.0.00.bz2.run --noexec --target /home/fliwa/temp/mus/op
Creating directory /home/fliwa/temp/mus/op
Verifying archive integrity... All good.
Uncompressing 4820 TOSHIBA MONITOR OPERATIONAL PACKAGE.....
fliwa-suse:/home/fliwa/Desktop/Latest_OP_UP # ./4820_21x_51x_Touch_FW_Update-1.0.00.bz2.run --noexec --target /home/fliwa/temp/mus/up
Creating directory /home/fliwa/temp/mus/up
Verifying archive integrity... All good.
Uncompressing 4820 TOSHIBA MONITOR UPGRADE PACKAGE.....
fliwa-suse:/home/fliwa/Desktop/Latest_OP_UP # ls -l /home/fliwa/temp/mus/op/
total 804
-rwxr-xr-x 1 root root 4987 Dec 5 18:00 installrpms
-rwxr-xr-x 1 root root 5753 Dec 5 14:55 MUS_UnInstall
-rw-r--r-- 1 root root 599828 Nov 26 11:47 toshiba-dspdrv-suse-1.2.1-1.0.i586.rpm
-rw-r--r-- 1 root root 192992 Dec 10 16:15 toshiba-monitor_upgrade-1.0.0-1.0.i586.rpm
fliwa-suse:/home/fliwa/Desktop/Latest_OP_UP # ls -l /home/fliwa/temp/mus/up/
total 4
drwxr-xr-x 3 root root 4096 Dec 4 05:37 4820
fliwa-suse:/home/fliwa/Desktop/Latest_OP_UP # ls -l /home/fliwa/temp/mus/up/4820/
pre-update stage task/
fliwa-suse:/home/fliwa/Desktop/Latest_OP_UP # ls -l /home/fliwa/temp/mus/up/4820/task/
conf/ fw/
fliwa-suse:/home/fliwa/Desktop/Latest_OP_UP # ls -l /home/fliwa/temp/mus/up/4820/task/fw/
total 64
-rw-r--r-- 1 root root 59478 Nov 28 11:15 -0459-00-MCS.HEX
fliwa-suse:/home/fliwa/Desktop/Latest_OP_UP # ls -l /home/fliwa/temp/mus/up/4820/
pre-update stage task/
fliwa-suse:/home/fliwa/Desktop/Latest_OP_UP # ls -l /home/fliwa/temp/mus/up/4820/task/conf/
total 4
-rw-r--r-- 1 root root 343 Dec 10 16:31 updatetask.conf
```

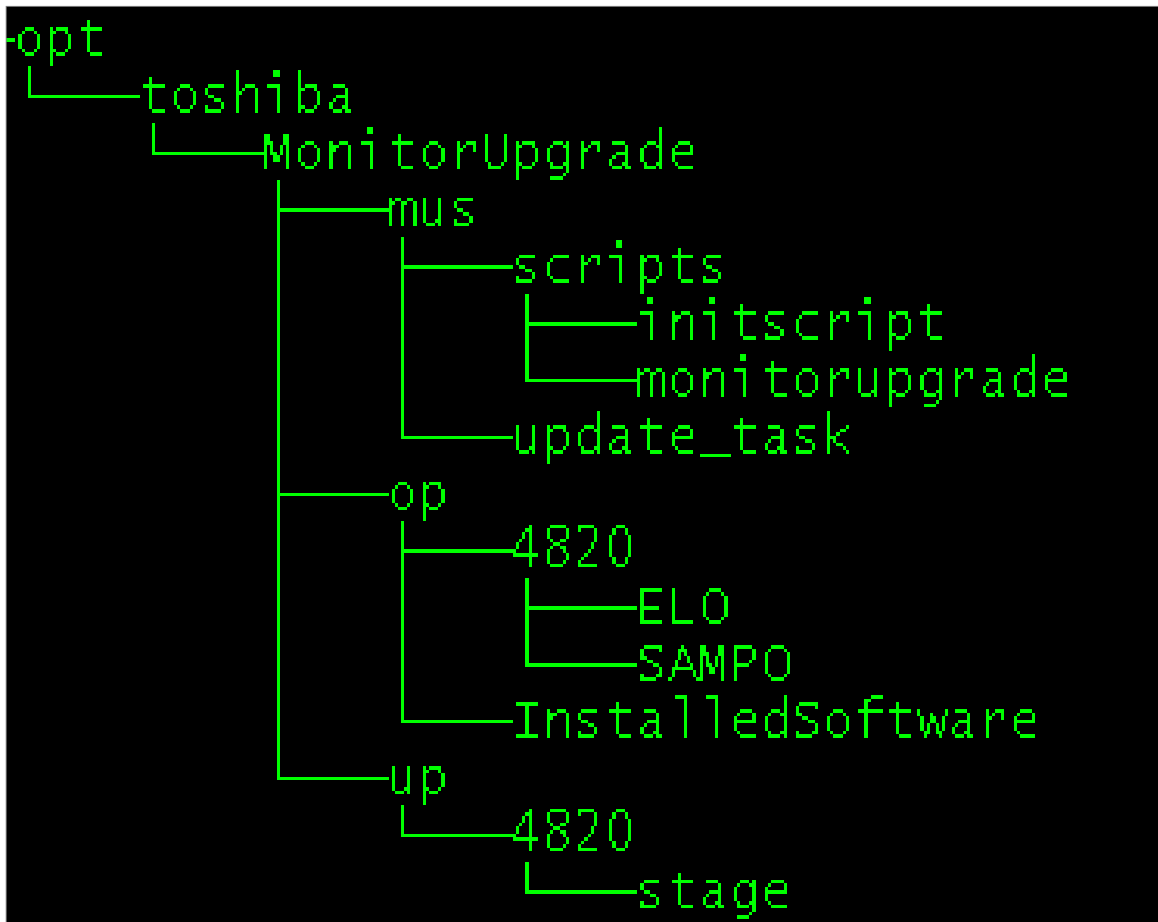
4 MUS Directory Structure

The MUS files can be found in three different areas:

1. MUS programs/scripts
2. Configuration
3. Log files

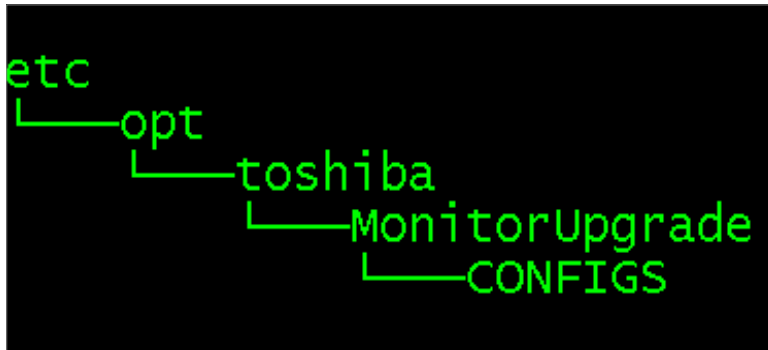
4.1 MUS base directory

The base MUS installation is under the *'/opt/toshiba/MonitorUpgrade/'* directory. The programs and scripts are stored in this location.



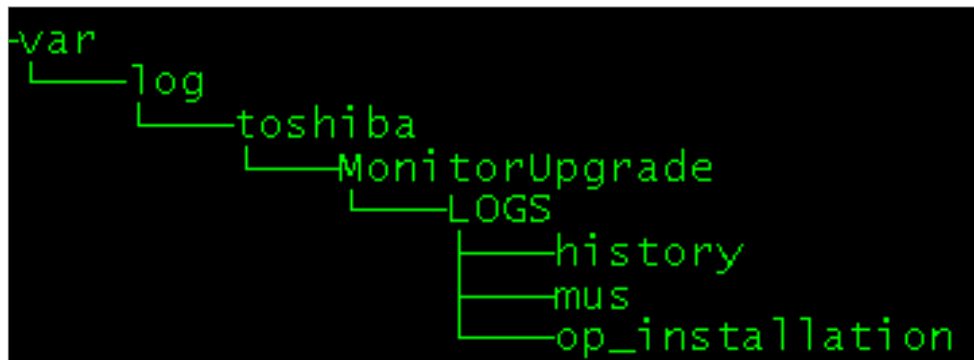
4.2 Configurations directory

The configuration files used by the MUS is in the *'/etc/opt/toshiba/MonitorUpgrade/CONFIGS/'* directory.



4.3 Logs Directory

The log files from the running of the MUS can be found under the *'/var/log/toshiba/MonitorUpgrade/LOGS/'* directory.



5 Frequently Ask Questions (FAQ)

1. What is inside the Operational Package?

Answer: Toshiba 4820 Operational Package contains the following components :

- Toshiba Monitor Inventory driver RPM
- Toshiba Monitor Upgrade RPM
- Video FW Update flash Utility
- Touch FW Update flash Utility

2. How do I check what is inside my package?

Answer: As root run this command: `./<Package.bz2.run> --list`

3. How do I know if MUS is installed?

Answer: Check if the service is running using the command: `sudo chkconfig MonitorUpgradeSvc`

4. I have an Upgrade Package (UP), what do I do with it?

Answer: As root run this command: `./<UpgradePackage.bz2.run>`

This will command will put an update task configuration file and firmware update file to their respective directories as listed in [section 4. MUS Directory structure](#).

5. Where can I find the log files of the MUS?

Answer: [Refer to section 4.3](#)

6. How do I uninstall the MUS?

Answer: Run the following command as root: `/opt/toshiba/UNINSTALL/MUS_Uninstall`

7. Can I just extract the contents of the package only?

Answer: Yes. Refer to section 3.7

8. When does the Update package take effect?

Answer: After running the self-extracting Upgrade Package (UP), the system needs to be rebooted in order for the MUS to execute the update task at hand.

6 Known Limitations

1. Only ONE Update task should be performed at a time. If there are multiple monitor types to be updated, then the user should run each update package for each monitor type one at a time in order to update all monitors.
2. Toshiba MUS will not support a customer setup where there is a HID-based UPS or similar devices which might be affected by the loading and unloading of `usbhid` by the ELO scripts. ELO is the vendor that provides the Update Utilities to update the Touch firmware of the Monitor, which in turn Toshiba MUS calls during reboot automatically depending on the update task.
3. Toshiba MUS will not proceed to update any succeeding monitors when updating the video firmware of the first attached monitor goes into recovery mode. This is the behavior that the Video Firmware Update.
4. During Video firmware update of Multiple 4820_2xx/5xx Monitors, it is expected that the user should not login to X first unless all the monitor video firmware are already updated. It is a known limitation that MUS can allow the user to login into X after the first monitor has been updated already while the other monitors are still doing the update in the background.
5. Auto recovery of monitor updates will be one monitor at time. This means that when there are multiple monitors which have failed Firmware Update, then the user should perform recovery multiple times.
6. While doing a Firmware Update, the system will have to reboot twice. First is after the Firmware update Package (UP) is executed, customer is asked to reboot the system to trigger the update. Second, while in the middle of the boot process just after the MUS finished update it will do another reboot.

Take note also that whether the update was successful or not the system will undergo two reboots to put back the system into operating condition again.