2012

UpStage Server Installation Manual

Gavin Chan, Scott Riddell, Daniel Han

AUT UpStage Team 2012

10/2/2012

Version History

Version	Changes made	Date	Who
1.0	Initial Draft	02/10/2012	Gavin Chan, Scott Riddell
1.1	Completed Steps on UpStage Source folder, Create and Starting Server	02/10/2012	Gavin Chan
1.2	Checked and Proof Read	02/10/2012	Daniel Han, Scott Riddell
1.3	Cleaned up and changed cover page	03/10/2012	Gavin Chan
1.4	Added Introduction and cleaned up in terms of supervisor specifications	09/10/2012	Gavin Chan
1.5	Added Fresh Installation and Uninstall Instructions	09/10/2012	Gavin Chan

Introduction

This document provides a step by step instruction on how to do a fresh installation of the UpStage server on a Debian platform and to also provide details on how to uninstall the server.

Step 1: Upstage Source Folder

Firstly start up the terminal and login as root; this can be done by typing in *su* in the terminal and typing in your password or by simply starting up the root terminal and login with your password. Once you have logged in as root, locate the source folder of the UpStage Server and access the folder.

(Figure 1)

The linux commands to help you achieve this are:

Linux Commands	Description
ls	Lists the accessible directories and files that are currently in your path
cd (directory name)	Changes into the selected directory that is available to access (As shown in figure 1)
cd ..	Go back a previous directory (eg. upstage@upstage:~/Desktop/Upstage242$ cd .. upstage@upstage:~/Desktop)
cd + tab	Shortcut for typing in directory name (type in cd then type in the first two letters of the directory name and press tab)

Step 2: Fresh Installation

The next step is the fresh installation of the UpStage server in the source folder. Please note that if your source folder is a subversion repository, doing a fresh installation will remove the subversion repository aspect of the folder. Therefore it is recommended to have a backup subversion of the UpStage source folder. To install the Upstage server make sure you have the install.py file in your source folder (can check using the *ls* command) and once you do, type in *python install.py* as shown in Figure 2

(Figure 2)

Once the installation is complete, you will get a welcome message from UpStage and instructions on how to create and start the server (Figure 3) in which those instructions will be explained in the next step of this manual.

```
File  Edit  View  Terminal  Help
l/upstage/chownme.sh
copied: /home/upstage/Desktop/Upstage242/upstage 2.4.2/img2swf.py -to- /usr/loca
l/upstage/img2swf.py
copied: /home/upstage/Desktop/Upstage242/upstage 2.4.2/speaker.py -to- /usr/loca
l/upstage/speaker.py
copied: /home/upstage/Desktop/Upstage242/upstage 2.4.2/upstage-admin -to- /usr/l
ocal/upstage/upstage-admin
Copied: /home/upstage/Desktop/Upstage242/upstage 2.4.2/upstage-admin.conf -to- /
usr/local/etc/upstage/upstage-admin.conf
copied: /home/upstage/Desktop/Upstage242/upstage 2.4.2/upstage-admin.conf -to- /
usr/local/upstage/upstage-admin.conf
copied: /home/upstage/Desktop/Upstage242/upstage 2.4.2/upstage-backup -to- /etc/
cron.weekly/upstage-backup
copied: /home/upstage/Desktop/Upstage242/upstage 2.4.2/upstage-backup -to- /usr/
local/upstage/upstage-backup
copied: /home/upstage/Desktop/Upstage242/upstage 2.4.2/upstage-server -to- /usr/
local/upstage/upstage-server


Finalizing Setup.


ln: creating symbolic link `/usr/local/bin/chownme.sh': File exists
ln: creating symbolic link `/usr/local/bin/img2swf.py': File exists
ln: creating symbolic link `/usr/local/bin/speaker.py': File exists
ln: creating symbolic link `/usr/local/bin/upstage': File exists
ln: creating symbolic link `/usr/local/bin/upstage-admin': File exists
ln: creating symbolic link `/usr/local/bin/upstage-admin.conf': File exists
ln: creating symbolic link `/usr/local/bin/upstage-backup': File exists
ln: creating symbolic link `/usr/local/bin/upstage-server': File exists
ln: creating symbolic link `/usr/local/bin/upstage-backup': File exists
****************************************************************


Thank you for choosing to use UpStage!
Visit upstage.org.nz for more information on UpStage


****************************************************************


To Create a new server run as root: upstage-admin create
To Start a server run as root: upstage-admin start servername
To See if any servers are active run as root: upstage-admin ls
AUT upstage development team (2011)


****************************************************************
root@upstage:/home/upstage/Desktop/Upstage242/upstage 2.4.2# 
    upstage@upstage: ~
```
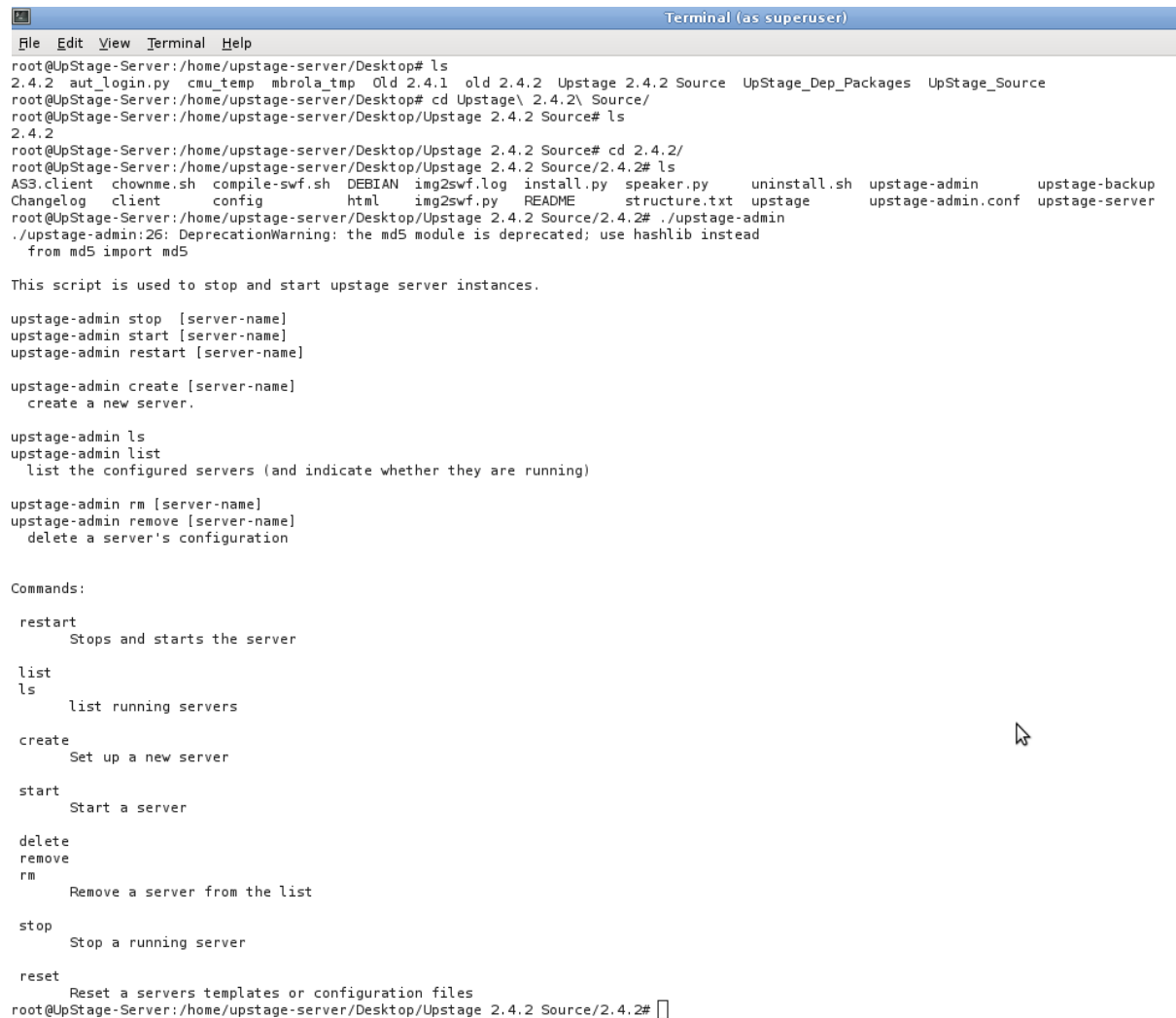
(Figure 3)

Step 3: Creating the Server

In the source folder of the UpStage server type in *upstage- admin* to display a list of commands for the server instance as shown below in Figure 4.

```
                                    Terminal (as superuser)
File  Edit  View  Terminal  Help
root@UpStage-Server:/home/upstage-server/Desktop# ls
2.4.2  aut_login.py  cmu_temp  mbrola_tmp  Old 2.4.1  old 2.4.2  Upstage 2.4.2 Source  UpStage_Dep_Packages  UpStage_Source
root@UpStage-Server:/home/upstage-server/Desktop# cd Upstage\ 2.4.2\ Source/
root@UpStage-Server:/home/upstage-server/Desktop/Upstage 2.4.2 Source# ls
2.4.2
root@UpStage-Server:/home/upstage-server/Desktop/Upstage 2.4.2 Source# cd 2.4.2/
root@UpStage-Server:/home/upstage-server/Desktop/Upstage 2.4.2 Source/2.4.2# ls
AS3.client  chownme.sh  compile-swf.sh  DEBIAN  img2swf.log  install.py  speaker.py    uninstall.sh  upstage-admin       upstage-backup
Changelog   client      config          html    img2swf.py   README      structure.txt  upstage       upstage-admin.conf  upstage-server
root@UpStage-Server:/home/upstage-server/Desktop/Upstage 2.4.2 Source/2.4.2# ./upstage-admin
./upstage-admin:26: DeprecationWarning: the md5 module is deprecated; use hashlib instead
  from md5 import md5

This script is used to stop and start upstage server instances.

upstage-admin stop  [server-name]
upstage-admin start [server-name]
upstage-admin restart [server-name]

upstage-admin create [server-name]
  create a new server.

upstage-admin ls
upstage-admin list
  list the configured servers (and indicate whether they are running)

upstage-admin rm [server-name]
upstage-admin remove [server-name]
  delete a server's configuration


Commands:

 restart
       Stops and starts the server

 list
 ls
       list running servers

 create
       Set up a new server

 start
       Start a server

 delete
 remove
 rm
       Remove a server from the list

 stop
       Stop a running server

 reset
       Reset a servers templates or configuration files
root@UpStage-Server:/home/upstage-server/Desktop/Upstage 2.4.2 Source/2.4.2#
```

(Figure 4)

To create a server, enter the create command with a thought up server name "*upstage-admin create (a server name)* " in which for the example in Figure 5 the server name is "testStage".

The terminal will then ask for a port number for the web server to listen on and provide a default number. You can change to an alternative port number by entering the number and pressing enter.

Next the terminal will ask for the port number for the flash sockets and policy files, the location for the logs, server data and process identifier. These should all be default unless suggested otherwise.

The terminal will then ask to set up an admin account and ask for a username and the admin password. Enter a username and the password twice to confirm (Note that the password will not be visible as shown below in Figure 5).

The server is then created.

```
 reset
        Reset a servers templates or configuration files
root@UpStage-Server:/home/upstage-server/Desktop/Upstage 2.4.2 Source/2.4.2# ./upstage-admin create testStage
./upstage-admin:26: DeprecationWarning: the md5 module is deprecated; use hashlib instead
   from md5 import md5
Setting up a new server
Which port should the web server listen on? (1024-65534) [8083]
Which port should the flash sockets use? (1024-65534) [7232]
Which port should policy files be served over? (1024-65534) [3000]
Where should the logs go? [/var/local/log/upstage/upstage-testStage.log] [/var/local/log/upstage/upstage-testStage.log]
Where should the server's data go? [/usr/local/share/upstage/testStage] [/usr/local/share/upstage/testStage]
Where should the process identifier (PID) file go? [/var/local/run/upstage/upstage-testStage.pid] (just press enter) [/var/local/run/upstage/
making directory /usr/local/share/upstage/testStage
You'll need to set up an admin user for this server (you can add more later)
what is the admin's username? []admin
enter the admin password
the same password againNOTE: these files and directories need to be writeable by the user that will be running the server:
/var/local/log/upstage/upstage-testStage.log
/var/local/run/upstage/upstage-testStage.pid
/usr/local/share/upstage/testStage
Let me chown the UpStage folders for you!
root@UpStage-Server:/home/upstage-server/Desktop/Upstage 2.4.2 Source/2.4.2# 
```
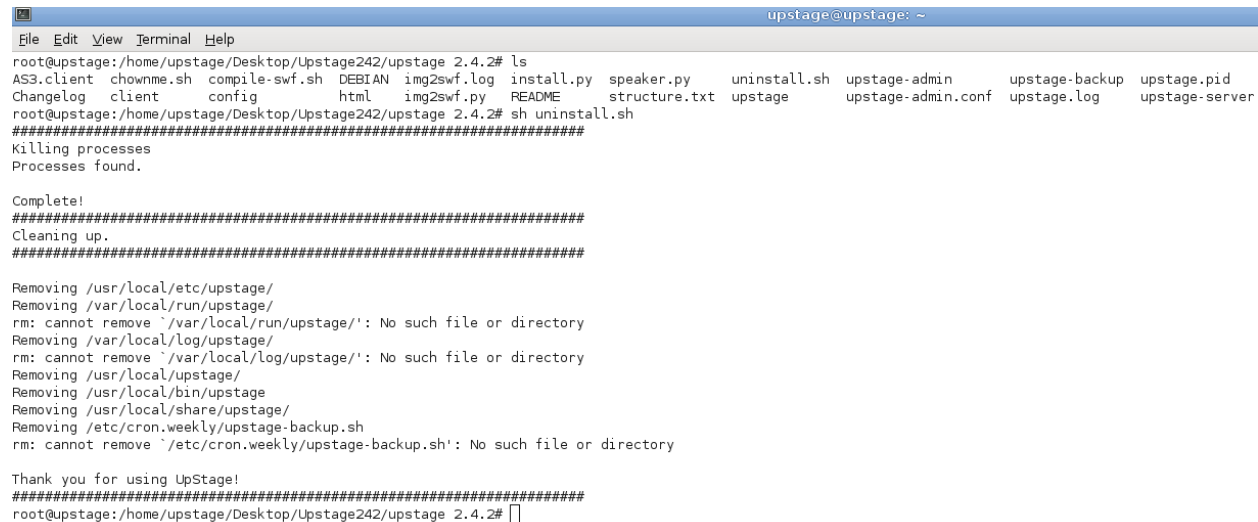
(Figure 5)

Step 4 Starting the Server

Once the server is created, use the command "*upstage-admin start (server name)* "as shown below in Figure 6 to start the server. The server should now be running and the website will be displayed when entered the chosen port number of the web server.

```
Let me chown the UpStage folders for you!
root@UpStage-Server:/home/upstage-server/Desktop/Upstage 2.4.2 Source/2.4.2# ls
AS3.client  chowme.sh  compile-swf.sh  DEBIAN  img2swf.log  install.py  speaker.py    uninstall.sh  upstage-admin       upstage-backup
Changelog   client     config          html    img2swf.py   README      structure.txt  upstage       upstage-admin.conf  upstage-server
root@UpStage-Server:/home/upstage-server/Desktop/Upstage 2.4.2 Source/2.4.2#  upstage-admin start testStage
./upstage-admin:26: DeprecationWarning: the md5 module is deprecated; use hashlib instead
   from md5 import md5
upstage-server -p 7232 -w 8083 -o 3000 -l /var/local/log/upstage/upstage-testStage.log -P /var/local/run/upstage/upstage-testStage.pid -b /usr/local/share/upstage/testStage
creating daemon
root@UpStage-Server:/home/upstage-server/Desktop/Upstage 2.4.2 Source/2.4.2# 
```

(Figure 6)

Uninstalling the UpStage Server

To uninstall the UpStage server, access the UpStage source folder and check if you have the uninstall.sh file, it is recommended to do a backup of the libraries and content of upstage before attempting to uninstall the upstage server. Once you have the uninstall.sh file, type in *sh uninstall.sh* to remove all local UpStage files and processes as shown in Figure 7.


```
                                                  upstage@upstage: ~
File  Edit  View  Terminal  Help
root@upstage:/home/upstage/Desktop/Upstage242/upstage 2.4.2# ls
AS3.client  chownme.sh  compile-swf.sh  DEBIAN  img2swf.log  install.py  speaker.py    uninstall.sh  upstage-admin       upstage-backup  upstage.pid
Changelog   client      config          html    img2swf.py   README      structure.txt  upstage      upstage-admin.conf  upstage.log     upstage-server
root@upstage:/home/upstage/Desktop/Upstage242/upstage 2.4.2# sh uninstall.sh
################################################################
Killing processes
Processes found.

Complete!
################################################################
Cleaning up.
################################################################

Removing /usr/local/etc/upstage/
Removing /var/local/run/upstage/
rm: cannot remove `/var/local/run/upstage/': No such file or directory
Removing /var/local/log/upstage/
rm: cannot remove `/var/local/log/upstage/': No such file or directory
Removing /usr/local/upstage/
Removing /usr/local/bin/upstage
Removing /usr/local/share/upstage/
Removing /etc/cron.weekly/upstage-backup.sh
rm: cannot remove `/etc/cron.weekly/upstage-backup.sh': No such file or directory

Thank you for using UpStage!
################################################################
root@upstage:/home/upstage/Desktop/Upstage242/upstage 2.4.2#
```

(Figure 7)