

MacPatch – Install & Setup Guide



Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

Table of Contents

MacPatch System Requirements	3
MacPatch System Tuning	4
MacPatch Database	5
Building the MacPatch Server	6
Apple Software Update Server	7
Server Configuration	7
JAVA Check.....	7
Server Configuration Steps	7
Download Content	8
Building the MacPatch Client Software	10
MacPatch Web Admin Setup	11
Backup.....	14
Backing Up The File System.....	14
Backing Up The Database.....	14

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

MacPatch System Requirements

Client Requirements:

- Mac OS X or Mac OS X Server 10.6.8 or higher
- Using Intel Hardware, PPC is not supported

Server Requirements:

- Mac OS X or Mac OS X Server 10.7 or higher
- Using Intel Hardware, PPC is not supported
- 4 GB of RAM, 8 GB is recommended
- Java v1.6 or higher
- MySQL version 5.1 or higher, MySQL 5.5 is recommended.
- Xcode 4.3 or higher

Server Ports Used:

- Default
 - 80, 443, 2600, 2601, 2602
- Configurable
 - 2600, 2601, 2602

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

MacPatch System Tuning

In order to get the most out of Mac OS X Client/Server operating system some of it's default limits need to be increased. Please run the following commands via Terminal application. Note, a reboot is recommended after applying these changes.

Please note, these adjusted settings are for a system with 4 Gigs of RAM.

```
# echo 'kern.maxfiles=20480' | sudo tee -a /etc/sysctl.conf
# echo -e 'limit maxfiles 8192 20480\nlimit maxproc 1000 2500' | sudo tee -a /etc/launchd.conf
# echo 'ulimit -n 4096' | sudo tee -a /etc/profile
```

Suggested Links:

- <http://support.apple.com/kb/HT4022>
- <http://support.apple.com/kb/HT3854>

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

MacPatch Database

The MacPatch environment requires a MySQL database to function. It is recommended that you install the database software prior to installing the MacPatch server software. It is recommended that MySQL version 5.5.x or 5.6.x Enterprise edition be used, but the community edition will also work. It is also recommended that the MySQL server be dedicated to the MacPatch server(s).

A MySQL configuration file (my.cnf) has been included with the MacPatch source as a starting baseline. Please note, the configuration file is for a MacPatch server containing the MacPatch database with 4Gig's of RAM.

Login as **root**

- `mysql -u root -p`

Run the following MySQL commands

```
mysql> CREATE DATABASE MacPatchDB;
mysql> CREATE USER 'mpdbadm'@'%' IDENTIFIED BY 'Password';
mysql> GRANT ALL ON MacPatchDB.* TO 'mpdbadm'@'%' IDENTIFIED BY 'Password';
mysql> GRANT ALL PRIVILEGES ON MacPatchDB.* TO 'mpdbadm'@'localhost' IDENTIFIED BY 'Password';
mysql> SET GLOBAL log_bin_trust_function_creators = 1;
mysql> FLUSH PRIVILEGES;
```

Delete MySQL anonymous accounts

```
mysql> DROP USER ''@'localhost';
mysql> DROP USER ''@'host_name';
mysql> quit
```

Load Database Schema

```
% mysql MacPatchDB -u mpdbadm -p < /Library/MacPatch/Server/conf/Database/MacPatchDB_tables.sql
% mysql MacPatchDB -u mpdbadm -p < /Library/MacPatch/Server/conf/Database/MacPatchDB_views.sql
```

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

Building the MacPatch Server

Building the base server software is very simple process. First the a tmeporary directory to hold the cloned MacPatch repository needs to be created. Clone the repository and run the "MPBuildServer.sh" script.

Please note, Xcode needs to be installed and the command line tools need to be installed as well. Xcode is a free download via the "App Store.app".

Install Steps via command line:

- `mkdir -p /Library/MacPatch/tmp`
- `cd /Library/MacPatch/tmp`
- `git clone https://github.com/SMSG-MAC-DEV/MacPatch.git`
- `/Library/MacPatch/tmp/MacPatch/scripts/MPBuildServer.sh`

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

Apple Software Update Server

The MacPatch server environment can be used with or without setting up your own Apple SoftwareUpdate server. By default all Apple Software Updates will be downloaded from Apple's servers.

If you choose to run your own Apple SoftwareUpdate server it is recommended to install it on a separate server. Also, if you plan on enabling and disabling patches/updates on the Apple SoftwareUpdate server you need to change the CatalogURL's in the "gov.llnl.mp.patchloader.plist" on the MacPatch server.

Server Configuration

The master MacPatch server requires that several components be setup and configured for the environment to work properly.

JAVA Check

Before any of the services are configured and launched now is the right time to make sure JAVA is installed. The best way to do this is to run " java -version" from the Terminal.app. If it's not installed the OS will prompt you to install it.

Server Configuration Steps

Run the following shell scripts in order to configure

1) DataBaseLDAPSetup.sh (Master)

- a. The database setup is required for MacPatch to function
- b. The LDAP/AD External Web Admin Authentication is optional, see note below about certificates.

Add Remote Certificates

If you have configured MacPatch to do user authentication via Active Directory/LDAP, which uses SSL to communicate, you might need to add the remote certificate to a list of trusted certificates. If a major certificate authority issued your certificate for your LDAP environment you can most likely skip this step.

Open a new Terminal window and run the following command

```
# /Library/MacPatch/Server/conf/scripts/addRemoteCert.sh -c "server.name.com 3269/636"
```

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

Now close the window and continue with starting the MacPatch server services.

2) WebAdminSetup.sh (Master)

- a. The MacPatch admin web console is required to use MacPatch. This section is an option for those who wish to setup additional servers for large environments.

3) WebServicesSetup.sh (Master)

- a. The MacPatch web services are required to use MacPatch.
Note: The TCP port used in this config is used for NON-SSL traffic.

4) PatchLoaderSetup.sh

- a. MacPatch requires gathering all of Apple Software updates from an Apple Software Update server. So that Apple patches can be assigned to a patch group for patching.

5) SymantecAntivirusSetup.sh (Optional)

- a. MacPatch supports patching Symantec Antivirus definitions. Not all sites use SAV/SEP so this step is optional.

6) StartServices.sh

- a. Depending on your choices this script will start all MacPatch services.

Download Content

Apple Content

Apple patch content will download eventually on it's cycle but for the first time it's recommended to download it manually.

Run the following command via the Terminal.app

```
# sudo -s _appserver /Library/MacPatch/Server/bin/MPPatchLoader -c \
/Library/MacPatch/Server/conf/etc/gov.llnl.mp.patchloader.plist
```

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

Symantec AntiVirus Defs

If you have elected to deploy Symantec AntiVirus definitions via MacPatch then it's also recommended that you download the content manually for the first time.

Run the following command via the Terminal.app

```
# sudo -s _appserver /Library/MacPatch/Server/bin/MPSAVDownloader -p \  
/Library/MacPatch/Server/conf/etc/gov.llnl.mpavdl.plist
```

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

Building the MacPatch Client Software

The MacPatch Agent package is very simple to build. A simple build script which you can extend and add more error checking to has been provided; located in the "scripts" directory of the GITHUB cloned master branch.

Edit *mplInfo.ini*

NOTE: the "mplInfo.ini" file is used to process the packages when uploading the agent via the web admin console. Clients will be updated based on "**version,agent_version,build**" properties.

```
[agent]
version=1.0.0
agent_version=2.1.1
build=1
framework=1.0.0
osver=10.6.8
pkg=Base.pkg
```

```
[updater]
version=1.0.0
agent_version=2.1.1
build=1
framework=1.0.0
osver=10.6.8
pkg=Updater.pkg
```

To build the client simply edit the "MPBuildClient.sh" script and set the "GITROOT" and "BUILDDROOT" variables accordingly. Then run the script via the Terminal.app. Inside of the \$BUILDDROOT/Client/Combined directory you'll find the MacPatch agent installer "MPClientInstall.pkg". While the installer is fully compiled it's not ready to be installed.

To prepare the agent for installation you'll need to login to the MacPatch Web Admin console and upload the newly compiled agent.

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

MacPatch Web Admin Setup

While all of the MacPatch server software is now installed a few configuration steps need to be completed before you may begin to roll out the agent to your clients.

The default user name is "**mpadmin**" and the password is "***mpadmin***"

1. Master Server Configuration
2. Proxy Server Configuration
3. Default Agent Configuration
4. Upload Client Agent
5. SoftwareUpdate Servers
6. Patch Groups
7. Software Distribution Groups

1. *Configure MacPatch Server Info*

- a. Go to Menu "Admin > Server > MacPatch Servers"
- b. Click the "+" icon and add a new record.
Note: If it's the first server it will automatically set it as master.

2. *Create New Agent Configuration*

- a. Go to Menu "Admin > Client Agents > Configure"
- b. Click the "Create New Agent Config" button
- c. Name the first configuration "Default"
- d. Set the following 3 properties to be enforced
 - i. MPServerAddress
 - ii. MPServerPort
 - iii. MPServerSSL
- e. Click the save button.
- f. Click the icon in the "Default" column for the default configuration. (**Important Step**)

3. *Upload Client Agent*

- a. Go to Menu "Admin > Client Agents > Upload and Deploy"
- b. In the "New Agent Upload" section click "Choose File"
 - i. **Base Agent PATH**
- c. Click the in the "Active" column for the "MPUpdateClient.pkg" row and change it to "Yes" and hit the enter key.

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

- d. Click the in the "Active" column for the "MPBaseClient.pkg" row and change it to "Yes" and hit the enter key.

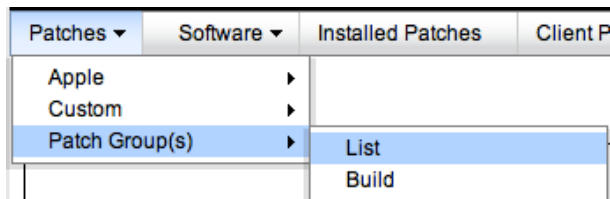
4. Create Client Agent Update Filter

- a. Go to Menu "Admin > Client Agents > Upload and Deploy"
- b. In the "Client Agent Update Filter" section click the "+" icon to add a new filter
- c. Fill in the needed values based on the screen shot below

The screenshot shows a dialog box titled "Add Record" with a close button (X) in the top right corner. It contains four input fields: "Attribute" with a dropdown menu showing "All", "Operator" with a dropdown menu showing "Equal", "Filter Value" with a text box containing "All", and "Condition" with a dropdown menu showing "None". At the bottom of the dialog are two buttons: "Submit" and "Cancel".

5. Create New Patch Group

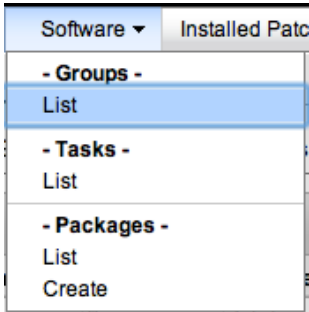
- a. Go to Menu "Patches > Patch Group(s)> Build"



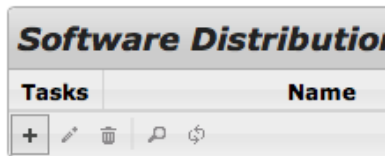
- b. Name the group "RecommendedPatches"
- c. Set group type to "Production"
- d. Select any available "Production" patches
- e. Click the "Create Group" button.

6. Create New Software Distribution Group

- a. Go to Menu "Software > - Groups - List"



- b. Click the "+" button to add a new group.



- c. Set group name to "Default"

 A screenshot of an 'Add Record' dialog box. It has a title bar with 'Add Record' and a close button (X). The dialog contains four input fields: 'Name' with the value 'Default', 'Description' with the value 'Default Software Group', 'Owner' (empty), and 'Modify Date' (empty). At the bottom, there are two buttons: 'Submit' and 'Cancel'.

- d. Setting the description is optional but advised.
- e. Click the "Submit" button.

Backup

Backup in MacPatch is very simple process. There are only two parts that need to be backed up. The two components are the files on the “Master” server and the database.

Backing Up The File System

For quick recovery Apple’s Time Machine software is a great solution. If your using a third party backup solution the main directory “/Library/MacPatch” is the only directory which **needs** to be backed up. The same directory needs to be backed up on additional distribution and proxy server.

Backing Up The Database

The MacPatch database backup procedure is a very simple one. Only the tables and their data need to be backed up.

If you have setup MySQL on a separate host and you have not installed the MySQL software the MySQL Workbench software is required as it contains the necessary binaries.

Requirements

- mysql and mysqldump
- MySQL Workbench software, installed in “/Applications”
(<http://www.mysql.com/downloads/workbench/>)

MySQL Workbench Setup

- Open Terminal, run the following commands
In -s /Applications/MySQLWorkbench.app/Contents/MacOS/mysql /usr/bin/mysql
In -s /Applications/MySQLWorkbench.app/Contents/MacOS/mysqldump /usr/bin/mysqldump

Backup Command

Script is located at:
/Library/MacPatch/Server/conf/scripts

Command:
php MPDBBackup.php <dbserver> <port> <Database> <user> <password> <dumpFileDirectory>

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM

Example:

php MPDBBackup.php MyDBServer.com 3306 MacPatchDB mpdbadm MyPass /private/tmp

Version 2.1

Rev 3

Modified On: 7/17/13 11:29 AM