

Maintain Multiple Mac OS X Machines with Radmin

Version 0.9.2

This document describes how to install, configure and use the radmin client and server tools to maintain a small lab of machines running the default Mac OS X install. The methodology described can also be applied to using radmin on other platforms. To work through the tutorial, you will need a server, a client and preferably a second client. This version of the tutorial has been written for use with Mac OS X 10.2, radmin version 0.9.2 and radmin assistant version 0.8.0.

1 - Server – Install and Configure

The radmin client can connect to a radmin server running on any supported platform. These directions describe how to install the server on a Macintosh running Mac OS X. Mac OS X server is not required to run radmin.

1. Download and install the radmin tools from <http://radmin.org>.

To install the server tools you will need to perform a customized install. During the install process, click the "Customize" button, select the three server packages and press install.

2- Client – Prepare the Base Load Set

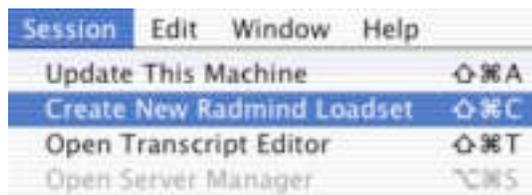
Prior to installing and using the radmin tools, you need to create a model machine that will be used to create the base load. A base load contains the operating system and other common files that every client will have. For this tutorial we suggest starting with a freshly installed Mac OS 10.2.2 machine.

1. Download and install the radmin tools from <http://radmin.org>

3 - Client – Create the Base Load

The Radmin Assistant provides a graphical user interface to the underlying radmin tools. It has been designed to simplify the radmin process by guiding you through each step.

1. Launch the Radmin Assistant found in /Applications/Utilities
1. By default, the Radmin Assistant begins a session to update the client. Since you are creating a new load, you need to change the session to "Create New Radmin Loadset. You can do so by selecting **Create New Radmin Loadset** from the **Session** menu.



3. Press **Continue** after reading the welcome message.

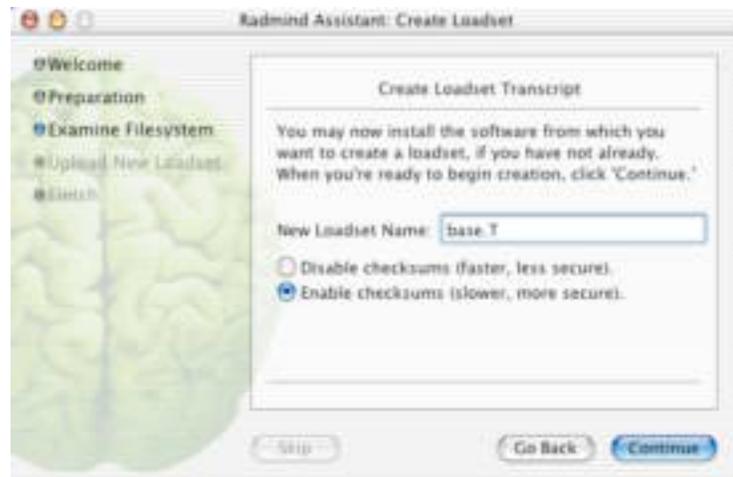


4. Since you are creating the initial load, you cannot use the server to update your machine. Press Skip to bypass this step.



5. During the “Examine File System,” step the Radmind Assistant creates a transcript that describes the base load. You can name your load by entering a name in the **New Loadset** field. For this exercise, enter **tutorial-base.T**.

Checksums can be used to verify the contents of a file and are useful in checking the integrity of a system. They take longer to calculate, but are highly recommended. Press **Continue** to create your base transcript. This may take a while depending on how large your base load is. Since the Radmind Assistant needs access to the entire file system, you will be asked to authenticate as an administrative user.



6. To look at the transcript in the Radmind Transcript Editor, press **Review Loadset Contents**
7. Taking a look at tutorial-base.T, you will notice that it is rather large. That is because it contains an entry for just about every file system object on your machine. When you are done reviewing the transcript, exit the Transcript Editor.
8. Enter the name or IP address of your radmind server where you wish to store the base load. For this exercise, leave checksums disabled. Press **Continue**. Uploading the new loadset may take a while depending on its size. Once the load set has been successfully stored on the server, you will be taken to the finish screen.



4 - Server - Verify the Loads and Create the Command File

We are now ready to verify the loads and make them available from the radmind server. Files and transcripts stored on the server are initially saved in /var/radmind/tmp/file/ and /var/radmind/tmp/transcript/ respectively. This is a staging area where loads can be verified before being made available via radmind. Files and transcripts available from the radmind server are stored in /var/radmind/file/ and /var/radmind/transcript/ respectively.

1. In the servers terminal window type:

```
[server:~] root# cd /var/radmind/tmp/transcript
```

This places you in the temporary storage area.

1. Before you make the loads available with the radmind server, you need to update tutorial-base.T to include checksums.

```
[server:~] root# /usr/local/bin/lcksum -c sha1 tutorial-base.T
```

lcksum reads the transcript tutorial-base.T and compares the checksum and file size listed in the transcript with the actual checksum and size of the corresponding file on the server. lcksum will update the transcript with the checksum and size of the file located on the server.

1. Now that both the base load has been verified, you are ready to make them available.

```
[server:~] root# cd /var/radmind/tmp  
[server:~] root# mv transcript/tutorial-base.T ../transcript  
[server:~] root# mv file/tutorial-base.T ../file
```

This moves the transcript and its associated files from the temporary directory where all loads are first stored into the working radmind directories.

1. Now you are ready to edit the command file to describe the load set consisting of the base load, tutorial-base.T and the negative overload, tutorial-negative.T.

```
[server:~] root# pico -w /var/radmind/command/apple.K
```

1. Add this line to the beginning of the command file:

```
p tutorial-base.T
```

The 'p' indicates that tutorial-base.T is a positive transcript.

Transcripts are listed in order of increasing precedence in a command file. The first transcript listed has the lowest precedence and the last the highest. If a file system object is listed in multiple transcripts, the transcript with the highest precedence is used.

For example, say you have a command file that looks like this:

```
p tutorial-base.T  
p overload.T  
n tutorial-negative.T
```

Where tutorial-base.T has /Applications listed as:

```
d /Applications          0775      0      80
```

And overload.T has /Application listed as:

```
d /Applications 0755 0 0
```

The ordering of the command file tells us that overload.T has a higher precedence, so it's listing for /Applications will be used. To guarantee that your negative transcript has the highest precedence, it should be listed last in the command file.

5 - Client – Test the Load Set

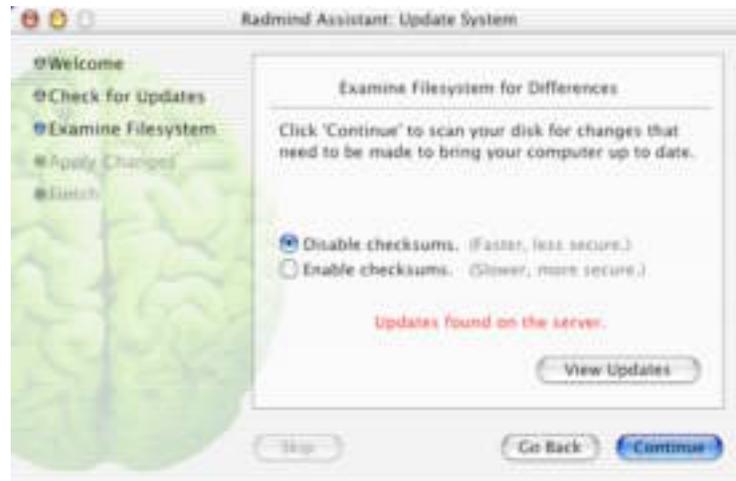
1. Launch the Radmin Assistant found in /Applications/Utilities
1. Press **Continue** after reading the welcome message.



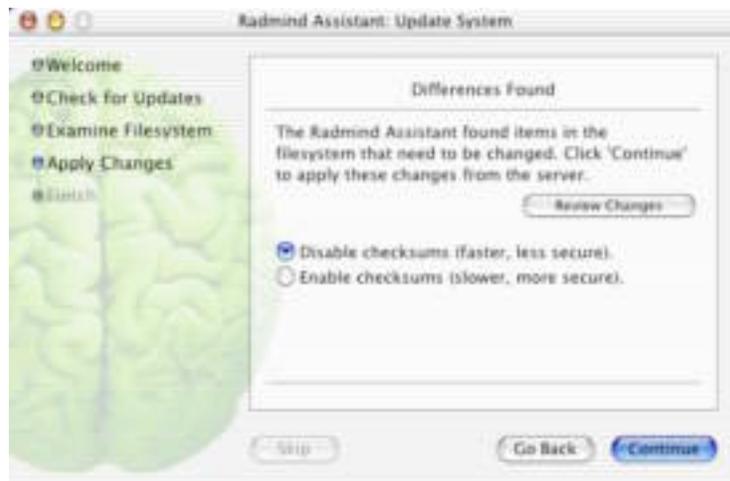
2. Enter the name of your radmin server that contains your base load. Press **Continue**. Authenticate as an administrative user when prompted. The Radmin Assistant will connect with the radmin server and verify that the client has the most up-to-date command file and related transcripts.



3. With the updated command file and transcripts, the Radmin Assistant can examine the file system to see if there are any differences. For more security you can use checksums, but for this exercise leave them disabled. Press **Continue**.



4. If there are any differences, the Radmin Assistant can bring the client up-to-date by applying the needed changes. Press the Review Changes button to see what changes will be made. To use checksums when downloading files from the server, enable checksums. For this exercise leave checksums disabled. Press **Continue**. This process could take a while depending on how many differences were found. To watch the changes being applied press **command-“L”** to open the log window. Press **command-“L”** again to close the log window.



5. Once all changes have been applied, the Radmin Assistant will tell you when the update is finished. Press **Finish** to end the session.

6 - Load Other Clients

Once you have tested the load, you can easily load other clients with the same load set in a few steps.

1. Download and install the radmind tools on the new client.
2. Launch the Radmind Assistant found in /Applications/Utilities
3. Press **Continue** after reading the welcome message.



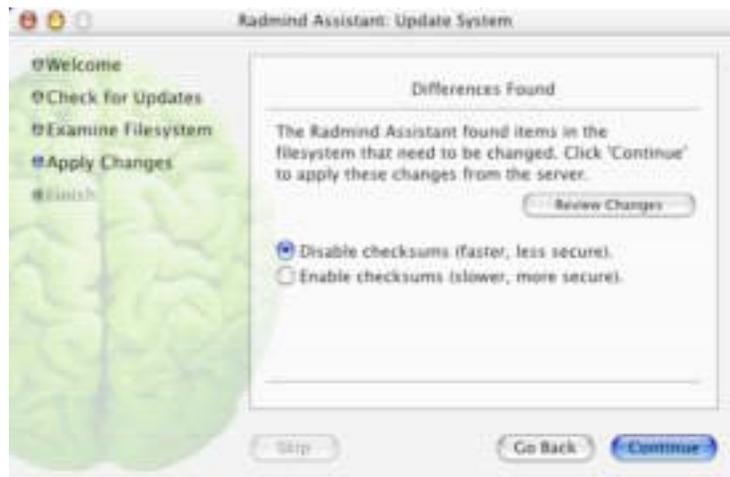
4. Enter the name of your radmind server that contains your base load. Press **Continue**.



5. Press **Continue** to examine the filesystem.



6. Press **Continue** to bring the client up-to-date. This process could take a while depending on how many differences were found.



Once all changes have been applied, the Radmin Assistant will tell you when the update is finished. Press **Finish** to end the session.

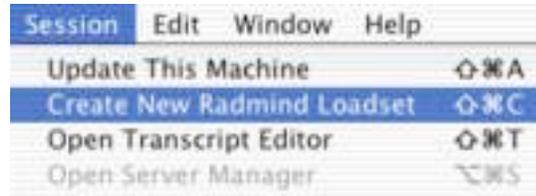
7 - Client - Adding New Software - Creating an Overload

With a working base load and negative transcript, you are now ready to create an overload. An overload can be used to add software to a client without storing it in the base load. This allows a system administrator to have a single base load used on every client and specialized overloads containing software and/or configurations shared among a group of clients. For instance, all clients with scanners might have an additional overload containing the scanning software and configurations.

In this example, we create an overload for the SFTP program fugu, but the same steps can be followed for any software.

1. Launch the Radmin Assistant found in /Applications/Utilities

- By default, the Radmind Assistant begins a session to update the client. Since you are creating a new load, you need to change the session to “Create New Radmind Loadset. You can do so by selecting **Create New Radmind Loadset** from the **Session** menu.



- Press **Continue** after reading the welcome message.



- Before creating the overload, you should update the client you are working on. Press update and follow the directions on bringing your client up-to-date.



- With your now up-to-date, you can install fugu.

6. During the “Examine File System,” step the Radmind Assistant creates a transcript that describes the overload. You can name your load by entering a name in the **New Loadset** field. For this exercise, enter **fugu.T**. Press **Continue** to create your overload transcript.



7. To look at the transcript in the Radmind Transcript Editor, press **Review Loadset Contents**
8. Taking a look at fugu.T, you might notice some files that are not fugu related. You can use the transcript editor to removed them from your overload. When you are done reviewing the transcript, exit the Transcript Editor.
9. Enter the name or IP address of your radmind server where you wish to store the base load. For this exercise, leave checksums disabled. Press **Continue**.



8 - Server - Checksum the Overload and Add It To the Command File

We are now ready to verify the overload and make it available from the radmind server.

1. In the servers terminal window type:

```
[server:~] root# cd /var/radmind/tmp/transcript
```

2. Add checksums to the overload.

```
[server:~] root# /usr/local/bin/lcksum -c sha1 fugu.T
```

3. Once overload has been verified, you can check it in.

```
[server:~] root# cd /var/radmind/tmp  
[server:~] root# mv transcript/fugu.T ../transcript  
[server:~] root# mv file/office.T ../radmind/file
```

4. Now you are ready to create edit the command file to include fugu as an overload.

```
[server:~] root# pico -w /var/radmind/command/tutorial.K
```

Add this lines to the beginning of the command file:

```
p    fugu.T
```

This indicates that any client using the tutorial command file should have fugu, the base load, and negative overloads.

9 - Client - Test the Overload

With the fugu overload successfully stored and a modified command file, we are ready to test the new load set.

1. Launch the Radmind Assistant found in /Applications/Utilities
2. Press **Continue** after reading the welcome message.



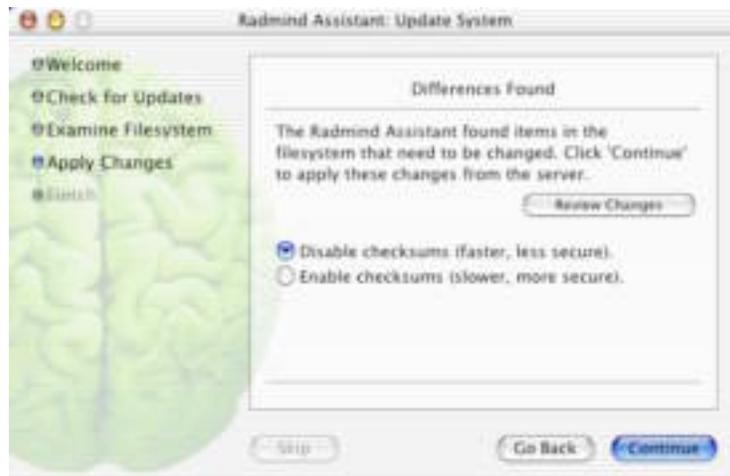
3. Enter the name of your radmind server that contains your base load. Press **Continue**.



4. Press **Continue** to examine the filesystem.



5. Press **Continue** to bring the client up-to-date. This process could take a while depending on how many differences were found.



Once all changes have been applied, the Radmind Assistant will tell you when the update is finished. Press **Finish** to end the session.

10- Server - Merge the Overload

In order to reduce the number of overloads listed in a clients command file, you can merge multiple overloads into one. For example, instead of having a base load with version 10.1 of the operating system and an overload for 10.1.1, 10.1.2, 10.1.3 and so on, you could merge the updates into the base load. The radmind server installation includes a tool that does this called lmerge.

In this example, we will merge two over loads that already exist on the server, fugu and tutorial-base.T.

1. On the server, move into the radmind transcript directory:

```
[server:~] root# cd /var/radmind/transcript
```

2. Use lmerge to combine the two over loads.

```
[server:~] root# /usr/local/bin/lmerge -f fugu.T tutorial-base.T
```

lmerge takes a list of transcripts in precedence order and created a new overload from them. In this case, the highest precedence transcript is office-update.T and the lowest precedence is office.T. The `-f` option causes lmerge to merge into tutorial-base.T instead of creating an entirely new overload.

lmerge writes the transcript line from each transcript to the new overload. If a file is listed in both transcripts, the transcript listing from office-update.T, the higher precedence transcript, will be written to the new overload.

Files are hard linked from their original location into the new overload to minimize disk usage. The next time a clinet updates, it will download the new tutorial-base.T that includes fugu.