

# Setting up a self-hosted Buzztouch 3.0.0 Control Panel on Amazon EC2's free tier

## Table of Contents

Why?	1
Prerequisites	1
Note	2
Creating the Instance	2
Putting a LAMP Stack on the Server Instance	7
Installing cURL on Your EC2 Instance	10
Getting the Buzztouch Files	10
Getting the Buzztouch Files to the EC2 Instance	12
Setting Up phpMyAdmin	15
Setting Up the Buzztouch Database	16
Setting Up Buzztouch on the EC2 Instance	18
Preparing Your Installation for API Key Validation	23
Validating Your API Key	26
What To Do After It's Working	27
Acknowledgements	27

## Why?

Buzztouch allows people with ideas but without much programming knowledge or experience to develop iOS and Android applications. Amazon Web Services's (AWS's) Elastic Compute Cloud (EC2) provides a free service tier for one year.

I was able to get my self-hosted Buzztouch Control Panel working on an Amazon EC2 instance that qualifies for the free tier. Along the way, I ran into several problems. Some I was able to solve myself and some required the help of veterans in the Buzztouch forums, but I did get everything resolved. This tutorial includes instructions specifically included to help its readers avoid the problems I had, so following this “how to” can save you some time and spare you some frustration.

The “how to” guide available for download from the Buzztouch web site appears to be a little bit outdated, as a few details have changed at EC2 and the Buzztouch Control Panel is now on version 3.0.0, and that document refers to “Buzztouch Server 2.0.” Still, that document is tremendously useful and was one of the main sources of information I used in installing my own self-hosted control panel and writing this updated guide. Links to that “how to” and two other documents I used appear in the acknowledgements section at the end of this document.

## Prerequisites

1. You must have an account to access Amazon Web Services. If you don't have one, follow the instructions at <http://aws.amazon.com/> to create one (“Sign Up”).
2. You must also be a Buzztouch member. Only members can self-host the Buzztouch Control Panel.

## Note

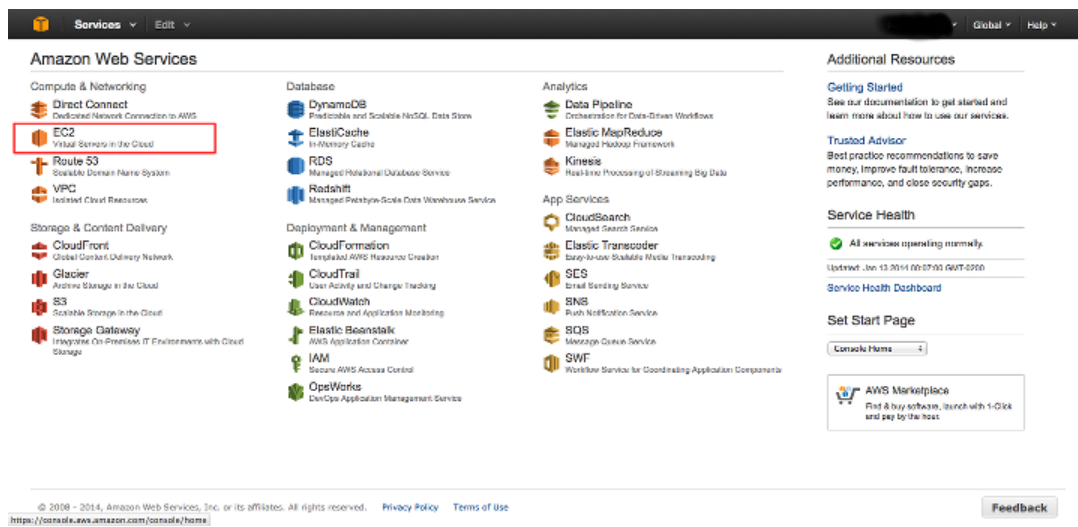
I am using OS X to do this. When I talk about opening a “Terminal,” that's the name for the command line interface in OS X. If you're using Linux, you'll have to open the corresponding tool on your system. Having lived a blissfully Microsoft-free life for six years and counting, I have no idea what a Windows user would have to do to be able to use SSH. If you're a Windows user and know how to run SSH from a command line, you can use the rest of this guide. If you're a Windows user and don't know how to run SSH from a command line, I can't give you step-by-step instructions like this “how to” guide, but with the help of a Windows guru, I might be able to help you with any problems you encounter

Let's get started. .

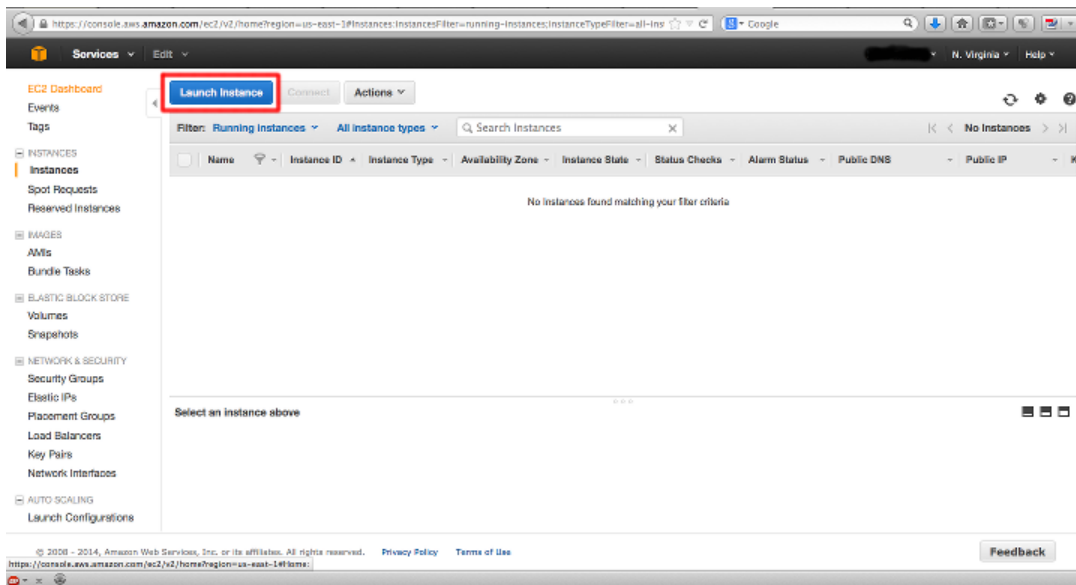
## Creating the Instance

You need to set up your server. In this tutorial, I'll go through setting up a 32-bit Ubuntu server with a complete LAMP stack. In the near future, I may try out some 64-bit operating systems to see if it makes a noticeable difference in the Buzztouch setup. I'll update this “how to” accordingly.

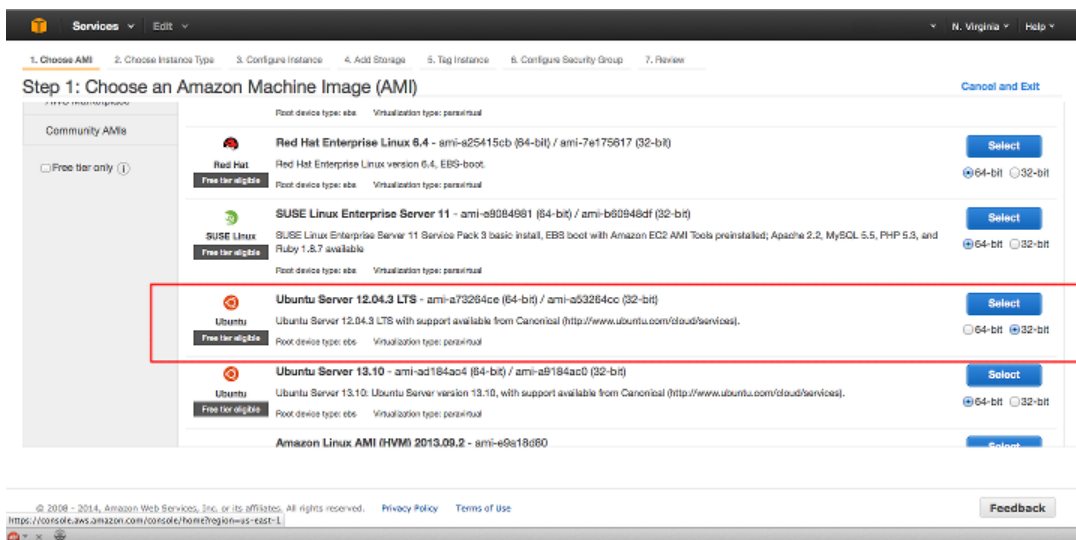
1. Log in to the AWS Management Console.
2. Click on “EC2” (see the red box in the image below).



3. Click on the blue “Launch Instance” button (see the red box in the image below).



4. Here I diverge a bit from the existing tutorial on setting up the Buzztouch Control Panel on EC2. Instead of selecting a specific AMI (Amazon Machine Image) that already has the LAMP stack installed, I will give instructions on how to construct one. I chose a more bare-bones AMI based on Ubuntu Server 12.04.3 LTS. I went with 32-bit. See the red box in the image below. I suspect the differences in the setup process will be quite small among different releases of different Linux distros, and there may be little or no difference if one chooses *64-bit* Ubuntu Server 12.04.3 LTS. Still, I am aware of some differences between Ubuntu and, for example, the Amazon Linux AMIs (different default users for SFTP, different commands for certain actions), so if you're not very comfortable and experienced with Linux, I suggest following my instructions as closely as possible, using the same operating system I did (32-bit Ubuntu Server 12.04.3).



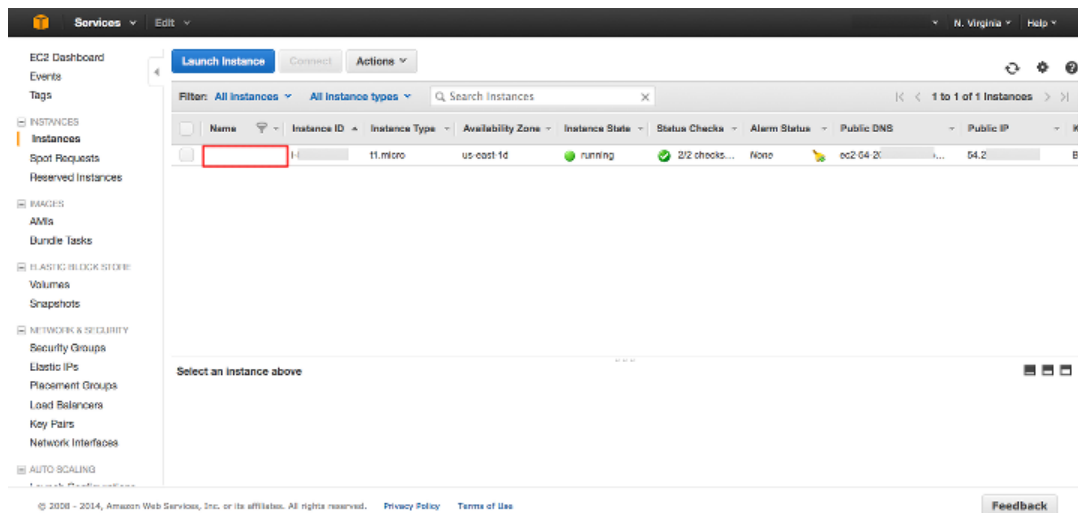
After choosing your Linux distro release number, choosing 32-bit or 64-bit, and clicking on the corresponding blue “Select” button, you’ll be taken to a screen on which you are asked to

choose an instance type.

5. The defaults on the “Choose Instance Type” page are fine, at least for now, as long as the instance type is “t1.micro,” which it should be. Click on the “Next: Configure Instance Details” button down near the lower-right corner of the screen.
6. That will take you to the “Configure Instance” page, where the defaults are also fine for now. Click on the “Next: Add Storage” button down near the lower-right corner of the screen.
7. That will take you to the “Add Storage” page, whose defaults are also fine for us for now. Click on the “Next: Tag Instance” button down near the lower-right corner of the screen.
8. That will take you to a page on which you can tag your new server instance. If it's your only instance, you don't need a tag at all. If you want to tag an instance to make it easier to find, you can do so, but for now this tutorial will instruct you to skip to the next screen. Click on the “Next: Configure Security Group” button near the lower-right corner of the screen.
9. On the “Configure Security Group” page, you need to allow certain kinds of network traffic to reach the machine. Specifically, in order to run Buzztouch, you'll need to have SSH and HTTP enabled, and possibly FTP, but FTP is not a secure protocol at all, while a client that can use SFTP (like FileZilla, for example) can make a secure connection with the server without the FTP network port needing to be open and transfer the files via SFTP (over SSH). The older “how to” says to enable FTP (and gives the port numbers for enabling SMTP, HTTPS, and a few others that aren't necessary for Buzztouch, but might be useful for other things). For now, let's not enable FTP. Having SSH enabled will allow us to use SFTP, which is better than my old, old friend FTP.  
A rule enabling SSH over port 22 is already set up by default when you arrive on the page. Click on the “Add Rule” button. From the “Protocol” pulldown, select “HTTP.” You don't need to change anything else.
10. To enable HTTPS (secure HTTP) on port 443, click on the “Add Rule” button again. This time, select “HTTPS” from the “Protocol” pulldown. You don't need to change anything else.
11. Click on the blue “Review and Launch” button near the lower-right corner of the screen. This will bring up the “Review Instance Launch” page.
12. Click on the blue “Launch” button near the lower-right corner of the screen. This will bring up the pop-up on which you're asked to create or specify a cryptographic key pair. In the older guide, the key pair was generated or specified earlier in the process. As the older how-to says, if you're not familiar with public key (asymmetric) cryptography, you should read up on this topic a little, especially to understand the importance of storing your private key securely.
13. In the pulldown on the upper part of the pop-up, select “Create a new key pair.” A text box will replace the lower pulldown that was on the pop-up. Make up a name for the key pair. I use something like “Buzztouch key pair 1,” but you can use just about any name you want. Put your name for the key pair in the text box and click on the “Download key pair” button on the pop-up.
14. Save the key pair to some place on your (local) computer. Don't forget where it is, because you will use this key pair instead of a password to log in to your EC2 server instance, and **there is no way to download another copy later**. If you lose the key pair (.pem) file, you won't be

able to access this EC2 server instance anymore and will have to create a new instance from scratch.

15. Once the key pair has been saved to your local computer, the blue “Launch Instances” button on the popup will be enabled. Go ahead and click on it. The screen will be updated as AWS launches and checks your new EC2 server instance. You should now have a server instance up and running!
16. There's a pulldown up in the upper-left corner of the screen called “Services ∨.” The inverted caret (“∨”) is the thing that tells you it's a pulldown. Click on the “Services ∨” pulldown and choose EC2. That will take you back to the Management Console.
17. Click on the “Instances” link on the left side panel, near the top. You should be taken to a page that looks like the image below (I've hidden my name, like in the previous images, and obfuscated some of the information about my server instance).

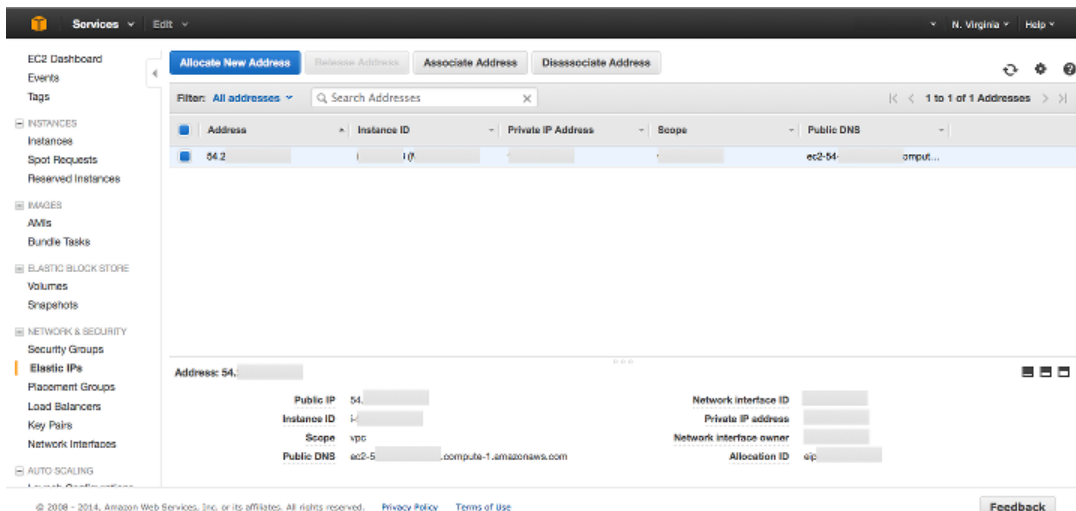


18. As the older guide points out, this is a good time to give your server instance a name. Hover over the “Name” field on the row of information about your instance. It should be blank now (see the red box in the image above). Click on the “Name” field and give your instance a name. You can be creative, but I used an informative name, so that if I have other instances in the future, I'll know why I created this one.
19. The “Instance State” field might not contain a green ball and the word “running” like in the image above right after you launch the instance, but within a few minutes of launching the instance, the state should be “running” and the green ball should appear.  
You're almost done setting up the instance. You just need to give it an IP address.
20. In the “Network and Security” group of options on the left side panel, click on “Elastic IPs.”
21. Click on the blue “Allocate New Address” button up near the top of the screen. See the image below the remaining instructions in this section.
22. A pop-up will ask if you are sure you want to allocate a new address. Click on the blue “Yes, Allocate” button. A new IP address will appear on the list. See the image below the remaining

instructions in this section.

23. Click on the empty checkbox next to the IP address on the list. It should be filled not with a check, but with a blue box. The “Release Address” and “Associate Address” buttons up near the top of the screen, just to the right of the blue “Allocate New Address” button, will be enabled. See the image below the remaining instructions in this section.
24. Click on the “Associate Address” button.
25. A pop-up will appear. Near the top, there is a pulldown entitled “Instance.” It should be set to “Select an instance.” Click on it and select your instance, which should include the name you gave it on the “Instances” screen.
26. Click on the blue “Associate” button. Your EC2 instance now has an IP address associated with it.
27. Write down the IP address. You'll need it in the next section and then several other times later.
28. If you've registered a domain and want it to point to this server (instance), you'll have to go to your domain registrar and make the name record point to this IP address. That's outside the scope of this document, and it's not necessary. The author of this document started without any registered domains.

NOTE: DETAILS OBFUSCATED IN THE IMAGE BELOW



Now that your instance is configured, running, and has an address, it's probably a good time to save an image. I'll remind you to do this again after you've got web server software running too.

29. For now, to get started, click on the “Instances” link near the top of the left panel in the Management Console.
30. Click on the empty checkbox next to your instance. It will be filled with a blue box indicating selection.
31. Go up to the “Actions ∨” pulldown button, click on it, and select “Create Image” from the drop-down menu that appears. A pop-up will appear.

32. In the “Image name” text field, enter a name (up to 128 characters) for your machine image. Note that this name can't be changed later, but you'll be creating other machine images later, so that's not too big a deal.
33. If you want to, enter a description (up to 255 characters) in the “Image description” text field.
34. The “No reboot” check box should be unchecked. Leave it that way. Click on the blue “Create Image” button in the lower-right corner of the popup. Your instance will be stopped, an image made, and then your instance will be restarted. In this process, snapshots of your storage units (think of them as disk drives) will be taken too. At this point, your instance should only have one 8 GiB storage unit associated with it.

## ***Putting a LAMP Stack on the Server Instance***

You have a server instance up and running in AWS's infrastructure, and you have its IP address. You ought to be able to connect to it via SSH. You enabled HTTP too, but for the moment, you don't have a web server running on our instance. In order to get one set up, you're going to use SSH and the cryptographic key pair you created and saved when you set up your instance to connect to the machine securely, both via the command line and using some secure file transfer software, and install and configure a complete LAMP stack (**L**inux operating system –already installed– + **A**pache web server + **M**ySQL database management system + **P**HP server-side scripting language for web pages). Let's get to work!

1. Open a Terminal (command line interface) and navigate to the directory (folder) where you saved the key pair (.pem file) in the previous section of this “how to” guide. Navigation in the command line involves the “cd” command. The details of where you have to “cd” depend on where you stored the .pem file.
2. When you're in the directory with the .pem file, whose file name I'll represent here with “[your\_key\_pair\_file\_name].pem,” run the following command in the Terminal to make sure only you have permission to use this key pair (otherwise, SSH on your instance might reject the key pair):

```
chmod 0600 [your_key_pair_file_name].pem
```

For example, if your key pair file name is mykeypair.pem, the command would be

```
chmod 0600 mykeypair.pem
```

3. Now that the permissions on the key pair file have been restricted, you can go ahead and try to log in to your instance via SSH by running the following command:

```
ssh ubuntu@[your_instance's_IP_address] -i [your_key_pair_file_name].pem
```

For example, if you called your key pair file mykeypair.pem, and your IP address is 999.999.999.999, the command would be

```
ssh ubuntu@999.999.999.999 -i mykeypair.pem
```

Note: if you have a domain name and have associated it with your instance's IP address, you can use that instead of the IP address in the `ssh` command.

4. You might be asked to confirm the identity of the server by typing “yes” in the command line. Go ahead and do so if that happens. You should then be connected to the instance – a secure



connection from your local computer to your server instance in the cloud! If not, the older guide suggests two possible reasons: either port 22 (SSH) isn't open (but since EC2 creates a rule allowing it by default, it should be open, unless you deleted that rule) or you've got a "bad instance." I've never had that experience, but the author of the older Buzztouch-on-EC2 guide writes "*You have port 22 open but the operation is still timing out. Sorry, looks like you have a bad instance. Try stopping it and starting it under the instance actions tab. If that doesn't work, terminate it in the instance actions and start over. That's one of the unfortunate things with Amazon. You can get a bad instance!*"

5. Now that you're logged in to the instance through SSL, become the "superuser" by entering the following command: `sudo -s`.  
If you don't see an error message, you are now the superuser (i.e., you're authorized to do a lot of stuff). The text before the prompt, instead of saying something like "`ubuntu@ip-xxx-xxx-xxx-xxx`" (with your instance's internal IP address on Amazon's network appearing in place of the `xxx-xxx-xxx-xxx` I put here), it will now say something like "`root@ip-xxx-xxx-xxx-xxx`."
6. Now that you're the superuser, make sure the apt-get package system is up to date by running the command `apt-get update`  
A whole bunch of information should appear in the Terminal, forcing it to scroll.
7. When apt-get is updated (i.e., when the information stops coming in the Terminal and the prompt and cursor appear), use it to install the LAMP server software by running the command (IMPORTANT: YOU MUST INCLUDE THE "^" AT THE END OF THE COMMAND, AS SHOWN IN THE LINE BELOW – I SUGGEST COPYING THE LINE AND PASTING IT INTO THE TERMINAL)  
`apt-get install lamp-server^`
8. You will be told how many megabytes of archives have to be retrieved, and how many additional megabytes of storage will be used on your instance if you continue. You will be asked if you want to continue, with the options "y" and "n." Type "y" and enter/return. The LAMP server software will be installed on your instance.
9. You will be prompted to enter a MySQL password. Choose one and do not forget it. You might want to write it down, or better, write down some mnemonic to help you remember it later without exposing the password to anyone who happens to see what you wrote. Installation and configuration will proceed automatically. This part could take a couple of minutes, but the Terminal is updated frequently with information about what's going on. You'll know it's done because text will stop appearing and you'll see the `root` prompt and a blinking cursor again.
10. Believe it or not, MySQL, Apache, and PHP are all now set up. If you want to check Apache, go to your management console and find the "Public DNS" web address for your instance. It should be something like `ec2-54-xxx-xxx-xxx.compute-1.amazonaws.com`, but with actual IP address numbers where I put `54-xxx-xxx-xxx`. Copy your instance's "Public DNS," open a new tab in your browser, and paste the "Public DNS" into the address field. The browser should display a plain white page with the text that appears in the following image:



# It works!

This is the default web page for this server.


The web server software is running but no content has been added, yet.

- To test PHP, you will need to create a new file. In the Terminal (where you're logged in to the EC2 instance as root), execute the command  

```
vi /var/www/testing.php
```

This will open the vi text editor. No holy wars, please. Prefer another text editor? Use it.
- Hit the “i” key to start inserting text in vi.
- Select the code on the line below with your mouse and copy it:  

```
<?php phpinfo(); ?>
```
- Right-click or control-click on the Terminal where you have vi running and paste the code into the new document. (Alternatively, you could type the code above directly into vi)
- Hit [esc] to stop inserting text.
- Hold [shift] and hit the Z key twice (to type two capital Z's in succession). The file will be saved and vi will quit.
- Open a new browser tab on your local machine. Paste your EC2 instance's “Public DNS” (something like `ec2-999-999-999-999.compute-1.amazonaws.com`) into the browser's address bar, and add the following to the end: `/testing.php`  
So if your instance's IP address were 999.999.999.999, you would have the following in your browser's address bar:  
`ec2-999-999-999-999.compute-1.amazonaws.com/testing.php`  
Once you have the right URL in the address bar, hit [enter]. You should see something like this:

PHP Version 5.3.10-1ubuntu3.9 	
System	Linux ip-172-31-45-179 3.2.0-54-virtual #82-Ubuntu SMP Tue Sep 10 20:49:45 UTC 2013 i686
Build Date	Dec 12 2013 04:06:44
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/mysql.ini, /etc/php5/apache2/conf.d/mysql.ini, /etc/php5/apache2/conf.d/pdo.ini, /etc/php5/apache2/conf.d/pdo_mysql.ini
PHP API	20090626
PHP Extension	20090626
Zend Extension	220090626
Zend Extension Build	API220090626.NTS
PHP Extension Build	API20090626.NTS
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
Registered PHP Streams	https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, tls
Registered Stream Filters	zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip, http.convert.iconv.*

You have a working LAMP stack on your EC2 instance.

## ***Installing cURL on Your EC2 Instance***

The Buzztouch Control Panel requires cURL to be installed. It's a relatively easy installation.

1. In a Terminal window, execute the following command on your EC2 instance:

```
apt-get install curl php5-curl  
The packages will be installed.
```

2. Restart Apache by executing the following command on your EC2 instance:

```
service apache2 restart
```

3. Check whether cURL is working by executing it on the default web page on your instance, which is the file in the physical location `/var/www/index.html`, by executing the following command on your EC2 instance:

```
curl http://127.0.0.1
```

If you see the contents of the simple “it works” default HTML file, cURL is working.

I just wanted to mention something here. I got as far as installing and configuring the Buzztouch control panel on an EC2 instance, but when I tried to validate my API key, it wouldn't work. I asked for help in the Buzztouch forums, and Niraj Shah replied and suggested I ask PSMDanny. Danny replied and made two immediate suggestions, one of which was to make sure cURL was installed and working on my server (instance). It wasn't. That's embarrassing, because the older Buzztouch-on-EC2 tutorial, which was one of my main sources as I was writing this updated tutorial, mentions that cURL needs to be installed and gives instructions on how to do so. On the other hand, other Buzztouch self-hosting guides don't mention installing cURL, which makes my mistake a bit less embarrassing.

## ***Getting the Buzztouch Files***

Now you've got an EC2 server instance running a complete LAMP stack. You're ready to get to the Buzztouch part of this “how to.” In this section, you'll download the files for your self-hosted Buzztouch Control Panel to your local computer and make a few simple changes to two of the files.

1. You need the Buzztouch self-hosting files. These can be obtained by opening a browser tab and going to the Self Hosted Software page in your Buzztouch account, at <https://www.buzztouch.com/account/selfHosted.php>
2. Under “Server API Keys” on the left part of this page, there is a list of API keys, which should have only one item. On the right side, there is a box entitled “Installation.” As of the moment in which this “how to” is being written, the box contains an image advertising a \$69.99 installation service, and under the images, a few links with brief text descriptions. One of the links says “download Buzztouch 3.0.0.” You should click on that link (or right-click or control-click and select “Save Link As...”). Save the zip file somewhere on your local computer. Leave the web page open, because you'll need some info from here a little bit later.
3. Unzip the contents of the zip file you just downloaded. That should give you a directory (folder) called `BT-server`. That directory contains a bunch of subdirectories and files.

In two of the files you just unzipped, there are occurrences of an incorrect version number, 2.1.9, where the correct version number, 3.0.0, should appear. I'll take you through the steps to change those now.

4. One of the two most important files is right in the `BT-server` folder. Open the `BT-server` folder and find the file called `config.php`. Open it with a text editor (like TextEdit, Notepad, gedit, vi, emacs, etc.).
5. In the text editor, go to the fifth line of `config.php`, where it should say  
`buzztouch Self Hosted v2.1.9`  
This won't affect anything visible, so you don't need to change it, but in the name of consistency, I recommend doing so. Change it to read  
`buzztouch Self Hosted v3.0.0`
6. Still in the text editor, navigate to line 72. Somewhere near there (in my case, I believe it was precisely line 72), you should see the following line of code:  
`define("APP_CURRENT_VERSION", "2.1.9");`
7. Go to that line of code and replace “2.1.9” with “3.0.0”  
The line should now read  
`define("APP_CURRENT_VERSION", "3.0.0");`
8. Save the file.
9. In the `BT-server` folder, find the folder called `install`. Open it.
10. In the `install` folder, find the file called `index.php`
11. The two modifications I'm going to tell you to make in this file are 100% aesthetic. They do not affect the functionality of the Buzztouch control panel in any way. They only affect what version the browser says is being installed when the Buzztouch installation page appears. That page may only be used once before you delete it from your instance (we'll take care of that when it's time), but since you may install Buzztouch again (on a new instance, or on multiple instances at once, or on a server with some other hosting provider, or even on a server in your home), you may see that page and what the title bar say when the page is open on multiple future occasions, and it's easy to avoid that with two minor changes now, so why not? To start, open `index.php` in the a editor.
12. The old version number, “2.1.9,” appears twice in `index.php`: once on line 212 and once on line 232. Replace both occurrences of “2.1.9” with “3.0.0”
13. Save the file.
14. There are similar 100% aesthetic changes to be made in the file `install_2.php` in the same folder (`.../BT-server/install`). The process is the same: open the file in a text editor and replace occurrences of “2.1.9” with “3.0.0” and then save the file. In `install_2.php`, “2.1.9” occurs twice. In my copy, it was on lines 108 and 128 of the file.
15. In `install_3.php`, “2.1.9” occurs three times. **IT IS IMPORTANT TO CHANGE ONE OF THEM!** The other two are “cosmetic.” The important one appears in or near line 90 (on mine it was line 90) of the file. The unmodified line reads  
`$configData .= "\ndefine(\"APP_CURRENT_VERSION\", \"2.1.9\");";`  
As usual, replace 2.1.9 with 3.0.0.

16. The two “cosmetic” changes in `install_3.php` are just like the ones in `index.php` and `install_2.php`. In my file, they appeared on lines 126 and 146. As in all the previous cases, replace 2.1.9 with 3.0.0.

17. Save the file.

18. **THIS NEXT MODIFICATION IS VERY IMPORTANT. WITHOUT IT, YOU WILL NOT BE ABLE TO VALIDATE YOUR API KEY AND USE YOUR SELF-HOSTED CONTROL PANEL IN AN EC2 INSTANCE.**

Open `../BT-server/admin/settings.php` in a text editor (or in an EDI like Xcode).

19. In `settings.php`, lines 62-64 should look like this:

```
if(isset($_SERVER["SERVER_NAME"])) $hostName = $_SERVER["SERVER_NAME"];  
$ipAddress = "";  
if(isset($_SERVER["SERVER_ADDR"])) $ipAddress = $_SERVER["SERVER_ADDR"];
```

Go to the line just below these, line 65, and insert the following line of code:

```
$ipAddress = "xxx.xxx.xxx.xxx";
```

with your EC2 instance's IP address in place of `xxx.xxx.xxx.xxx` as usual.

20. Save the file.

The first time I tried to set up my Control Panel on an EC2 instance, I uploaded the files without modifying them, and when I opened the Buzztouch installation page, I saw “v2.1.9” and asked about it in the Buzztouch forums. SmugWimp told me about the changes that needed to be made in `config.php`. That's important. But then I noticed that modifying `config.php` doesn't change what it says on the Buzztouch installation page. I determined what lines could be changed in `/install/index.php`, `/install/install_2.php`, and `/install/install_3.php` in order to make it say “3.0.0” on the installation pages and in the browser's title bar when those pages are open, and while doing so, discovered that there's a “2.1.9” in `install_3.php` that could make a difference. When I tried to validate my API key, I noticed that the IP address for my Buzztouch server (EC2 instance) was not a valid routable public IP address; it was a `172.xxx.xxx.xxx` address, its internal address on the Amazon network. I couldn't find a place to change it on the self-hosted pages, so the best kludge that came to mind was hard-coding my EC2 instance's public IP address into the settings page's `.php` file. PSMDanny later told me there probably isn't a better way. I put the instructions for how to make all the modifications to these files in this part of the “how to” guide, **before** the files are uploaded to the EC2 instance, so that you won't need to modify the files remotely, and if you keep the modified copy on your local computer, you won't need to modify those files ever again even if you have to build a new instance, change hosting providers, etc., requiring you to reinstall version 3.0.0 of the Buzztouch Control Panel.

## ***Getting the Buzztouch Files to the EC2 Instance***

I wanted to use a nice tool to get the Buzztouch files over to the instance, so I figured out how to do it using FileZilla, a file transfer program with a nice graphical interface. FileZilla supports SFTP (FTP over SSH on network port 22, so you don't need to enable the far-from-secure bare FTP protocol). So in this section, you'll download and install FileZilla, and then use FileZilla to put the Buzztouch files on the instance. The document `BuzzTouchServerInstall.pdf`, available from the Buzztouch web site, uses FileZilla too, and was one of the sources used in constructing this “how to.”

Because I'm unfamiliar with how to do things on recent versions of Windows, I haven't been giving Windows instructions, but there is a version of FileZilla for Windows. I can't guarantee the user interface options are exactly the same as those in the OS X version (or accessed the same way through the graphical interface), but there are versions of the FileZilla client software available for download for OS X, Windows, and Linux.

1. If you don't yet have FileZilla running on your local computer, go to the Filezilla web site at <https://filezilla-project.org/> and download the FileZilla **client** software for your local computer's operating system (the Filezilla server software only exists for Windows, and you won't be using it here).
2. Once you've downloaded the file, go ahead and install it.
3. Once FileZilla is installed, go ahead and launch it.

You have to use the private key you created when we set up your instance in order to log into the instance. You therefore need to let FileZilla know about the key pair.

4. Open the FileZilla preferences. On OS X, this is accomplished by opening "Preferences" from the "FileZilla" menu or by hitting the key combination ⌘,
5. In the "select page" box on the left side of the Preferences window, under "Connection," choose "SFTP." You might need to click on the triangle next to "Connection" in order to show "SFTP" and the other options in the "Connection" group first.
6. A list (it should be empty now) of private keys appears in the main part of the Preferences window. Under the (empty) list are two buttons: "Add keyfile..." and "Remove key." The "Remove key" button should be disabled (greyed out) if there are no keys currently in the list. Click on the "Add keyfile..." button and navigate to where you saved the key pair (.pem) file when you created the EC2 instance. Select the key pair (.pem) file and click on the blue file selection dialog's "Open" button.
7. A pop-up will appear and tell you that the key pair is not in a format supported by FileZilla. You will be given the option of converting the key to the right format. Accept that option by clicking on the blue "Yes" button in the pop-up.
8. A dialog will appear. You will need to specify a file name for the .ppk (PuTTY private key) file, which is where FileZilla will store the private key for future use. I left mine in the same place where I'd put the .pem file and gave it a very similar name. Put a filename in the "Save as" field up near the top of the dialog. As soon as you've put one or more characters in the "Save as" field, the "Save" button in the lower-right corner of the dialog should turn blue. Once you've got the filename you want specified for the .ppk file, click on the "Save" button to convert the key, save it in .ppk format, and add your key to the list on the SFTP page of the Preferences window.
9. Once the .ppk file has been saved and your private key appears in the list of keys, click on the blue "OK" button to close the Preferences window.
10. Now open the Site Manager in FileZilla. On OS X, this is done by selecting "Site Manager" from the "File" menu or by using the key combination ⌘S
11. Get the "Public DNS" web address from your Management Console. It should look something

like `ec2-xxx-xxx-xxx-xxx.compute-1.amazonaws.com`, but with your instance's IP address in place of `xxx-xxx-xxx-xxx`. Paste that web address into the “Host:” field on the FileZilla Site Manager window. Because you're going to use the standard SSH port, 22, you don't need to specify a port on the Site Manager.

12. Set the protocol to “SFTP – SSH File Transfer Protocol” in the “Protocol” pulldown.
13. Set the logon type to “Normal” in the “Logon Type” pulldown.
14. Set the user name to `root` in the “User” text field.
15. Leave all the other fields alone. Click on the blue “Connect” button.
16. You will be asked if you want to accept the key being used to identify the server. Accept it. You will then be logged in via SFTP to your EC2 instance. Its file structure from the `/root` directory will be shown in the right side of the FileZilla main window.
17. Double click on “.” in the list of files and folders in the right side of the lower part of the left-right separated part of the window. This represents “move up a directory.” You will be shown the contents of the `/` directory. Scroll down until you see the `/var` directory.
18. Double-click on the `/var` directory. When its subdirectories appear, scroll down until you see the `www` directory (`/var/www`).
19. On the left side, navigate to where the `BT-server` directory containing the Buzztouch software was put when you unzipped the `.zip` file you downloaded from Buzztouch. Click on the `BT-server` directory, hold the button, and drag the `BT-server` directory from wherever it is on the left side of the left-right split part of the window to the `www` directory on the right side of the split part of the window.  
  
The SFTP transfer of hundreds of files will begin (in my case, there were 548 files to transfer). The whole transfer takes a little time, but you can see the progress as it goes. Let it finish.
20. When the files have all been transferred from your local computer to the `/var/www` directory of your EC2 instance, you should see the `BT-server` directory in the `/var/www` directory of your EC2 instance (on the right side in FileZilla). Double-click on the `BT-server` folder.
21. You need to change permissions in the `/files` directory inside the `BT-server` directory (i.e., for `/var/www/BT-server/files`). You can do this directly in FileZilla. I told you it was a nice tool! Right-click or control-click on the `/files` directory and select “File permissions...” from the context menu. This will open the “Change File Attributes” window for the `/files` directory. If the numerical value is 755 (corresponding to the owner having read, write, and execute permission in the directory, while everyone else has read and execute permission, but NOT write permission), leave it that way. If the numerical value is anything other than 755, change it to 755 (and the permission checkboxes will change accordingly).
22. Click on the blue “OK” button to close the “Change File Attributes” window.
23. You're done with FileZilla for now, so you can go ahead and close it.

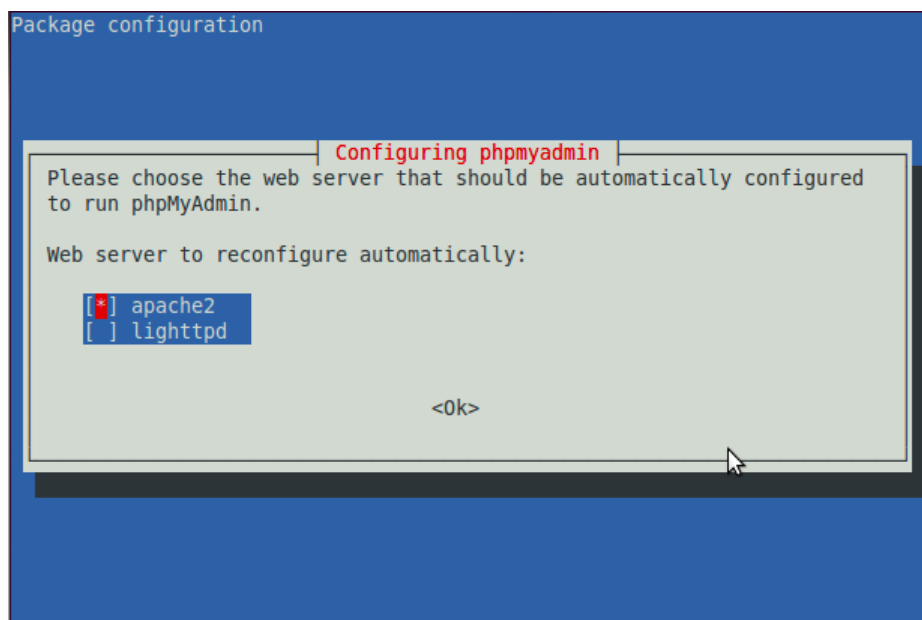
Now you have the Buzztouch self-hosting files in the `/var/www` directory of your EC2 instance with the right permissions. You're getting close! The next step will be to proceed with the actual installation of the Buzztouch self-hosted control panel.

Before you proceed with the Buzztouch installation, you need to create a MySQL database and some user accounts for that database.

## Setting Up phpMyAdmin

A very popular tool for administration of MySQL databases is phpMyAdmin. There are ways to use the command line (Terminal) to do the few things you need to do to set up Buzztouch (basically, set up an empty database and a user with certain privileges on that database), but it's pretty easy to install and use phpMyAdmin, and if you need to do any future database administration, phpMyAdmin will make that much easier. In this section, you'll install phpMyAdmin. In the next section, you'll use it to set up the Buzztouch database.

1. In the Terminal, run the following command:  
`sudo apt-get install phpmyadmin`  
The installation will begin. You will be told how much disk space will be used, and you will be asked if you want to continue.
2. Reply with “y” and [enter].
3. A screen will appear asking you for which web server you want to configure phpMyAdmin: apache2 or lighttpd. Make sure apache2 is selected (you can change the selection with the up and down arrow keys on your keyboard) and hit [enter]. See the image below.



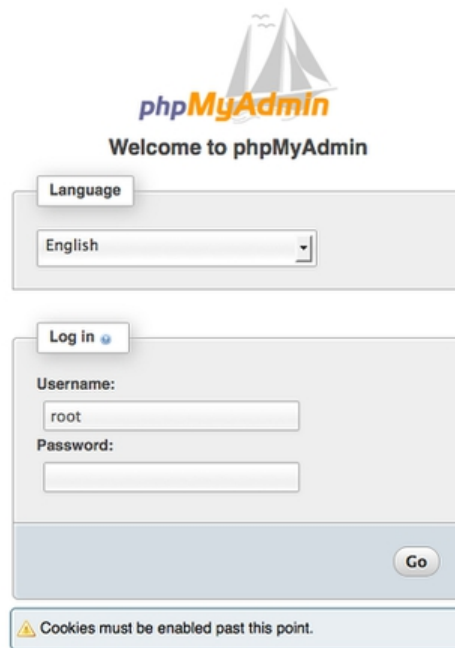
4. Another screen will appear, asking you if phpMyAdmin should configure a database for its own use. Make sure “yes” is selected (you can change the selection with the arrow keys on your keyboard) and hit [enter].
5. Yet another screen will appear, this one asking you for the password to the MySQL administrative account that should be used to create the database and its user. You should enter the MySQL root password you specified when you were installing the LAMP stack on your EC2 instance. When you've entered the password, hit [enter].



6. Another screen will appear, this one asking you to create a password for the application user account to be created for phpMyAdmin. Create a password by typing it in (followed by [enter]) and remember it. Write down the password or, better yet, some mnemonic that only you would understand.
7. A final screen will appear, this one asking you to confirm the password you just created. Type it in again and hit [enter]. The database will be created.
8. I found that phpMyAdmin was not installed where I would have expected it (in `/var/www` so I could get to it by putting `[my_instance's_URL]/phpmyadmin` in my browser's address bar). Instead, it was installed to `/usr/share/`. Since phpMyAdmin appears to be made up of PHP files, and those files only reference each other using relative location (e.g., “in this same directory where I live”), I thought I might be able to just move the phpmyadmin folder from `/usr/share` to `/var/www`, but I wasn't completely sure I could do that without breaking anything. The correct way to solve the problem once phpMyAdmin has been installed to `/usr/share` is to create a symbolic link (to the installation) in your `/var/www`. The command to do that is  

```
sudo ln -s /usr/share/phpmyadmin/ /var/www/phpmyadmin
```

If somebody tells me how to tell apt-get to install phpMyAdmin in `/var/www`, I'll update this “how to.”
9. Point your browser to `[Your_instance's_Public_DNS]/phpmyadmin`. You should see the phpMyAdmin login screen:



phpMyAdmin  
Welcome to phpMyAdmin

Language  
English

Log in

Username:  
root

Password:

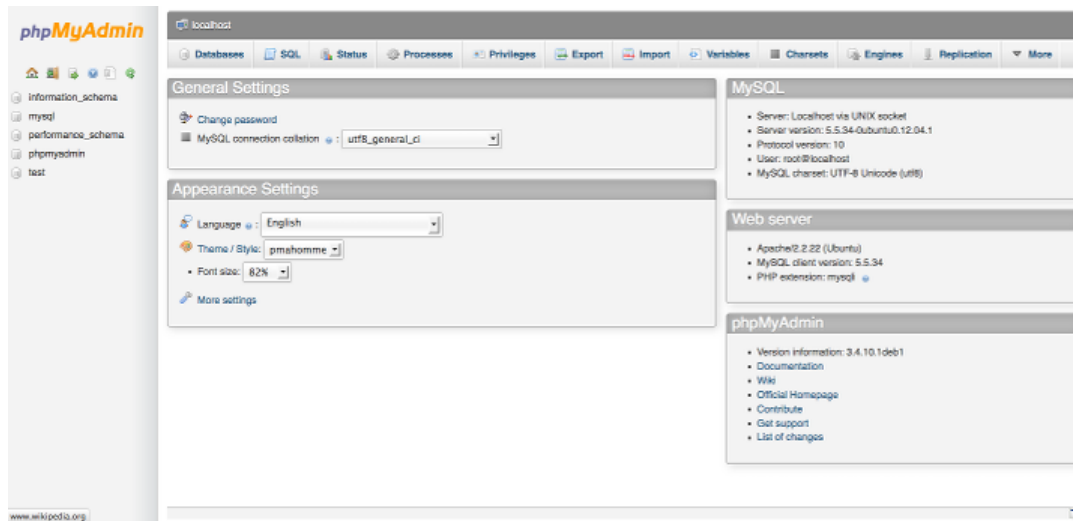
Go

⚠ Cookies must be enabled past this point.

## Setting Up the Buzztouch Database

Now that you've got phpMyAdmin installed on your EC2 instance, you can use it to set up the Buzztouch database.

1. Enter “root” in the “Username” text field.
2. In the “Password” text field, enter the MySQL root password you specified when you were installing the LAMP stack on your EC2 instance. Hit [enter] or click on the “Go” button. You will be taken to the phpMyAdmin main screen, which looks like the image below.



3. Click on the “Databases” tab, the first one in the upper-left corner.
4. Create a new database. To do this, enter a name for the database (something like BT or BTDB is recommended, because it's short and relatively informative – I used “BTDB,” and some tutorials suggest using “BT.”), choose “utf8\_unicode\_ci” in the “Collation” pulldown, and click on the “Create” button.

Now you need to create a user or two for the database. I created one with total DBA privileges, even though the MySQL root user already has those, because I might want to separate the BTDB administrator from the MySQL root user. I called that user `btodb_admin`. I also created a user account for the BTDB database, the one I intend to use with Buzztouch. I gave that user less access and power than the admin account. As soon as I find out the minimum privileges that the user account can have in order for the self-hosted Buzztouch Control Panel to work, I'll update this “how to” (and change the privileges on my database!). You can always change privileges later via phpMyAdmin. I will even restrict the BTDB admin account as far as I can without compromising the usability of Buzztouch on my EC2 instance.

5. It's time to create your database's admin account. Click on the “databases” tab.
6. Click on the name of your newly created Buzztouch database in the list of databases that appears. You'll notice that the name of the database you selected appears up at the top of the screen.
7. Click on the “Privileges” tab. You'll see a list of users that have privileges on your Buzztouch database. Click on “Add a new User” under the list.
8. You'll be taken to a screen on which you can specify the name, password, and privileges for the new user. Fill in a username (I used `btodb_admin`) and a password (twice – there's a “Re-type”

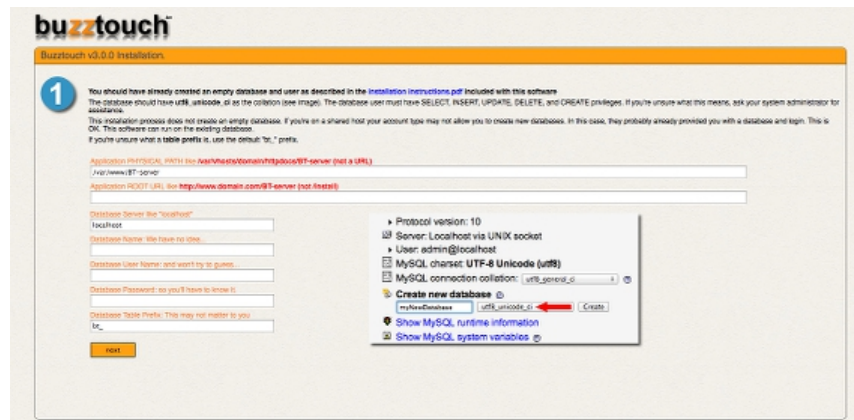
password confirmation field). Be sure you can remember the password. Write it down, or better yet, write down a mnemonic only you would understand. Don't type anything in the field next to the "Generate Password" button.

9. Under "Database for user," the option selected should be to grant the privileges specified below on your Buzztouch database (mine is called BTDB). Make sure that one is selected. Under "Global privileges," go ahead and tick all the checkboxes.
10. Click on the "Create User" button.
11. Click on the "Databases" tab again, and when the list of databases comes up, click on your Buzztouch database again.
12. Click once again on the "Privileges" tab. When the list of users appears, click on "Add a new User" (under the list) again.
13. This time, you'll create the user account Buzztouch will use. Give it a name (I used bt\_db\_user) and password (twice, including the "Retype" password confirmation field). Make sure the privileges are being granted on your Buzztouch database (mine is called BTDB). For now, go ahead and tick all the checkboxes in the "Data" and "Structure" sections under "Global privileges." Click on the "Create User" button.

## ***Setting Up Buzztouch on the EC2 Instance***

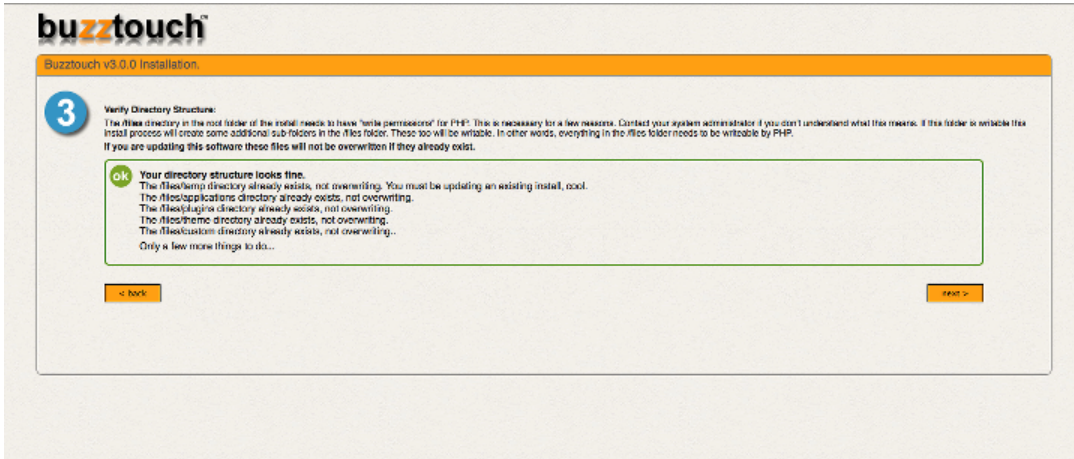
You've come a long way. To recap, you started by creating an EC2 instance (a "virtual server") with 32-bit Ubuntu 12.04.3 LTS as its operating system. You installed a complete LAMP stack on that instance. You tested Apache and PHP to make sure they were working. You installed cURL and php-curl and tested those. You installed phpMyAdmin and used it to set up the database and corresponding user account Buzztouch will use. Now you're finally ready to go through the actual installation of the Buzztouch Control Panel on the instance.

1. To start, open a new browser tab. You'll need your "Public DNS" web address again, the one that looks something like `ec2-xxx-xxx-xxx-xxx.compute-1.amazonaws.com`, but with your instance's IP address in place of `xxx-xxx-xxx-xxx`. You can always get it from the information about your instance on the "Instances" page of the Management Console at EC2. You're going to need to navigate to the `/www/BT-Server/install` directory in the browser. That means entering an address that looks something like `http://ec2-xxx-xxx-xxx-xxx.compute-1.amazonaws.com/BT-server/install`, but with your EC2 instance's IP address (with triplets separated by "-" instead of ".") in place of `xxx-xxx-xxx-xxx`. This should open the Buzztouch 3.0.0 installation page. This page has some instructions and a bunch of text fields where you'll have to supply important information.

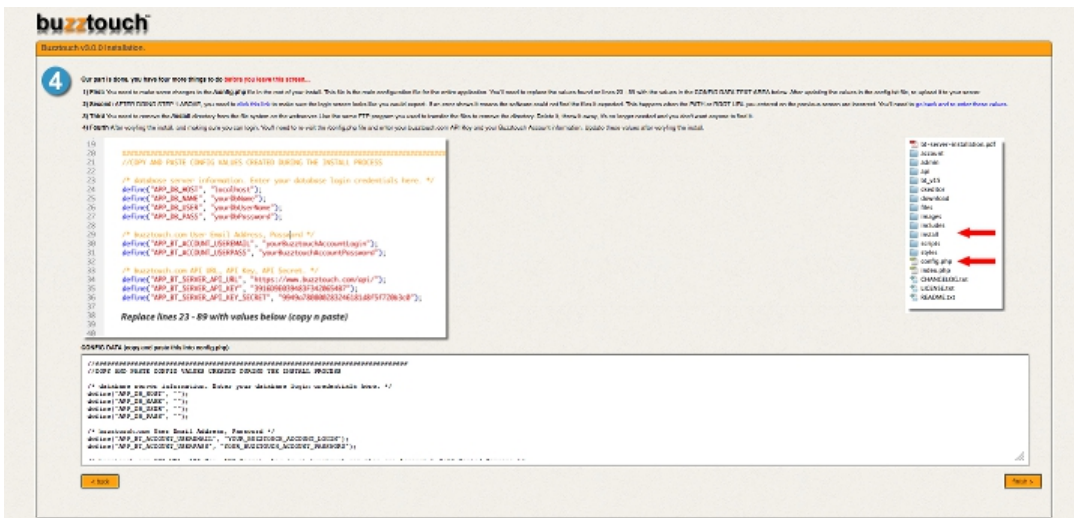


2. The page has seven text fields. The red text above each one describes pretty well what needs to be entered there.
  - 2.1. The “Application PHYSICAL PATH” field should already be completed. Leave it alone.
  - 2.2. In the “Application ROOT URL” field, enter the URL for your Buzztouch Control Panel. If you've been following my examples, it should be something like <http://ec2-xxx-xxx-xxx-xxx.compute-1.amazonaws.com/BT-server>, where, as usual, xxx-xxx-xxx-xxx should be replaced by the EC2 instance's IP address, with “-” instead of “.” separating the four numbers. If you have a domain and have pointed it at your EC2 instance, then you can enter something more like <http://mydomain.com/BT-server>, where your actual domain should replace “mydomain.com.”
  - 2.3. The “Database Server” field should have been automatically set to “localhost.” If it was, leave it.
  - 2.4. For the “Database Name,” “Username,” and “Password” fields, use the name of the empty database you created in phpMyAdmin, the username of the user you created in phpMyAdmin, and the password for that user, respectively.
  - 2.5. The “Database Table Prefix” field should have “bt\_” in it. Leave it like that.
  - 2.6. Hit the “Next” button.
3. If there was an error, go back and check your database name, username, and password. If you're stumped and your password contains special characters (like “\$,” for example), try modifying the password to remove the special characters in phpMyAdmin and make the corresponding changes in the appropriate fields on the Buzztouch 3.0.0 installation page.
4. Once you get the initial installation page's text fields filled in without errors, follow the instructions on the next page and make sure you see the image as described in the text on the page. If you don't see the check mark inside the red square, then your root URL is not pointing at your Buzztouch installation correctly.

- On the next page, you should get a message saying your directory structure is fine. If not, make sure the /files directory is where it's expected to be (.../BT-server/files) and make sure it has the right permissions (0755). This should be OK already, because we set the permissions in the section of this “how to” guide on getting the Buzztouch files to the EC2 instance.



- Follow the instructions on the next page that appears. There are four things that have to be done before you can leave this page.



Here are the instructions as they appear on that page (this tutorial won't follow them exactly, but I present them here anyway):

- First:** You need to make some changes to the /config.php file in the root of your install. This file is the main configuration file for the entire application. You'll need to replace the values found on lines 23 - 89 with the values in the CONFIG DATA TEXT-AREA below. After updating the values in the config.txt file, re-upload it to your server.
- Second:** AFTER DOING STEP 1 ABOVE, you need to [click this link](#) to make sure the login screen looks like you would expect. If an error shows it means the software could not find the files it expected. This happens when the PATH or ROOT URL you entered on the previous screen are incorrect. You'll need to [go back and re-enter these values](#).

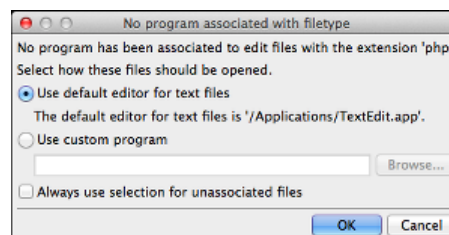
**3) Third** You need to remove the `/install` directory from the file system on the webserver. Use the same FTP program you used to transfer the files to remove the directory. Delete it, throw it away, it's no longer needed and you don't want anyone to find it.

**4) Fourth** After verifying the install, and making sure you can login. You'll need to re-visit the `/config.php` file and enter your buzztouch.com API Key and your Buzztouch Account information. Update these values after verifying the install.

What follows is my suggestion of how to complete your Buzztouch Control Panel installation on your EC2 instance.

6.1. Immediately below the instructions is a section of PHP code with color coding. Below that is a text box containing a bunch of (plain text) PHP code. You must replace some of the lines in `config.php` with the PHP code from the text box. Here are the details of how to do that.

- a) Copy the contents of the text box.
- b) In FileZilla, open the Site Manager (it's an item in the File menu). The only entry should be the one you created for your EC2 instance (and it logs in as `root`). It should be selected and the "Connect" button should be highlighted. Click on the "Connect" button to connect to your EC2 instance. The connection will be made using your key pair for authentication, so you won't need to enter a password.
- c) You'll see the FileZilla interface, of which the middle section is split in half. The left part shows the filesystem on your local machine, while the right part shows the filesystem on the remote server (in this case, your EC2 instance). On the right side, click on `..` to move up one directory to the `/` directory.
- d) Scroll down and open the `/var` directory by double-clicking on `var`.
- e) Scroll down and open the `/var/www` directory by double-clicking on `www`.
- f) Open the `/var/www/BT-server` directory by double-clicking on `BT-server`.
- g) In the lower-middle panel on the right-hand side of FileZilla, right-click (or control-click) on `config.php`. Only part of the name may appear, like `config.p...` From the context menu that appears, choose "View/Edit" (it should be the third item in the menu).
- h) A dialog will appear asking in what editor the file should be opened. I used the standard text editor for my local machine.



i) In the text editor you chose, remove the lines corresponding to the ones you just copied from the Buzztouch installation page. Basically, it's the lines from the one that reads  
`//COPY AND PASTE CONFIG VALUES CREATED DURING THE INSTALL PROCESS`

to the one that reads

```
//END COPY AND PASTE VALUES CREATED DURING THE INSTALL PROCESS
```

Replace those lines with the text copied from the Buzztouch installation page.

- j) Now you need to make a few more minor changes to the file before saving it again and closing it.

Find the section near the beginning of the file that contains some lines like the following:

```
//COPY AND PASTE CONFIG VALUES CREATED DURING THE INSTALL PROCESS
/* database server information. Enter your database login
credentials here. */
define("APP_DB_HOST", "");
define("APP_DB_NAME", "");
define("APP_DB_USER", "");
define("APP_DB_PASS", "");
```

Insert between the empty quotation marks in the APP\_DB\_HOST line the word localhost.

For the APP\_DB\_NAME, use the name of the database you created in phpMyAdmin. I called mine BTDB, so those two lines in the config.php on my EC2 instance look like this:

```
define("APP_DB_HOST", "localhost");
define("APP_DB_NAME", "BTDB");
```

For APP\_DB\_USER, enter the username of the database user you created in phpMyAdmin. I called mine btodb\_user. If my database user password were thePassword, then those two lines in my config.php would read

```
define("APP_DB_USER", "btodb_user");
define("APP_DB_PASS", "thePassword");
```

- k) Find the lines that look like this in config.php on your remote server (still editing in the text editor on your local machine):

```
/* buzztouch.com User Email Address, Password */
define("APP_BT_ACCOUNT_USEREMAIL", "");
define("APP_BT_ACCOUNT_USERPASS", "");
```

Enter the e-mail address you used to register at buzztouch.com between the quotation marks in the first of these two lines. Enter your Buzztouch password between the quotation marks in the second.

- l) There will be more changes to make to config.php shortly, but for now, you can save it with the latest modifications and close it. Go back to FileZilla, where a dialog will inform you that the file has been changed and ask if you want to upload the modified file. Click on the "OK" button to confirm this and update the file on your EC2 instance.

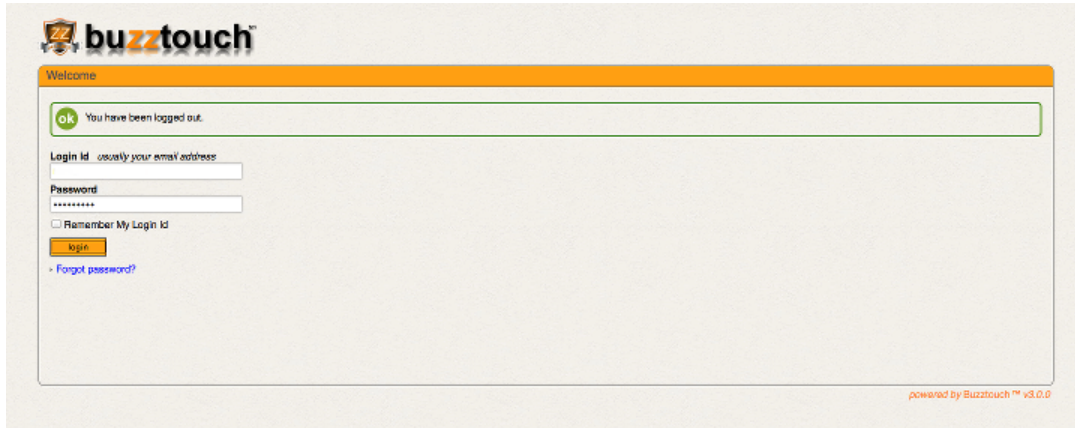
- 6.2. Go back to the browser tab in which you have the Buzztouch installation page open. In the second step on the on-screen instructions, click on "click this link"

If you see the Buzztouch login screen (see below), move on to the next step. If there's an error, you have to check the PATH and ROOT URL you entered on the previous screen. The easiest way to get there is to click on the "go back and re-enter these values" link in step 2 of the on-screen instructions.

- 6.3. Before doing the third step in the instructions on the Buzztouch installation page, I suggest checking that you can log in.



- a) Open your Control Panel login page in a new browser tab. The URL should be something like <http://ec2-xxx-xxx-xxx-xxx.compute-1.amazonaws.com/BT-server>, where, as usual, your EC2 instance's IP address, with “-” instead of “.” as separators, should replace “xxx-xxx-xxx-xxx”



- b) Log in here. In the “Login Id” field, use the address [fans@buzztouch.com](mailto:fans@buzztouch.com). In the “Password” field, enter the password [buzztouchpro](#). When you've logged in, immediately go and change your login information.
- Start by clicking on the “Update Email Address” link on the left side of your account home screen in the Buzztouch Control Panel. There are two fields in which you must enter your e-mail address. Use the e-mail address you used to register with Buzztouch and make sure it's entered correctly in both text fields. Click on the “submit” button.
  - Click on the “Update Password” link on the left side of the page. There are three text fields on the page that opens. In the first, enter the current password, [buzztouchpro](#). In the second, enter the password you want to use to log in to your self-hosted Buzztouch Control Panel. In the third, enter the same password again. Click on the “submit” button.
  - Finally, click on the “Update Your Name” link on the left side of the page. Enter your first name and last name in the text fields on the page that opens. Click on the “submit” button.
  - If you want to change your time zone, use the “Update Time Zone” link on the left side of the page.
- c) I've left the deletion of the `/install` directory for a later step.

## ***Preparing Your Installation for API Key Validation***

1. Open the Buzztouch web site in a browser tab and log in to your Buzztouch account.

- On your account page on the Buzztouch web site, click on the “Self Hosted Software” link on the left side.
- There should be two boxes on the page that opens. In the one on the left side, you should see a link for the license corresponding to your control panel. Click on it.

**Self Hosted Software**  
The Buzztouch Self Hosted software allows you to run a unique, branded, and customizable Buzztouch control panel on your own website. We offer a [Professional Installation Service](#) for those that are unable to install it themselves - It's technical.

**Server API Keys**

	# Req.	Allowed
Research Dept Control Panel	7	100

Click a server name to view details.  
Allowed is per month.

**Installation**

**Professional Installation**  
\$69.99 **Members Only**

[Learn More >](#)

Install the Buzztouch Self Hosted software on your own or have a Buzztouch Professional do it for you.

- Download Buzztouch v3.0.0 (2.7 mb)
- LAMP stack required (Linux, Apache, MySQL, PHP). Unix only. Windows servers not supported.
- [Self Hosted How To Guides](#)

- Information about your API key will appear.

(Some details have been obfuscated in this image – on your screen, you should see the server IP address filled in)

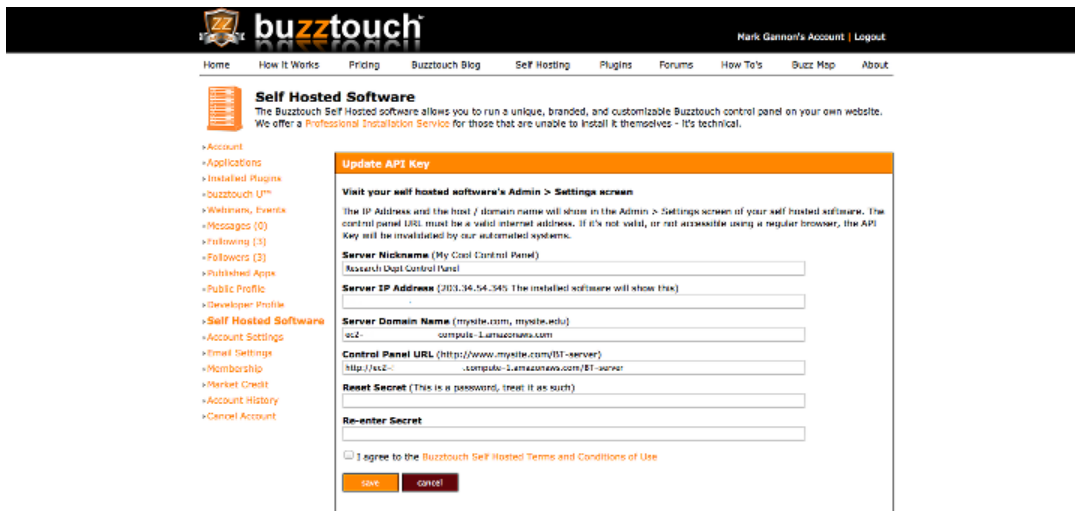
**Self Hosted Software**  
The Buzztouch Self Hosted software allows you to run a unique, branded, and customizable Buzztouch control panel on your own website. We offer a [Professional Installation Service](#) for those that are unable to install it themselves - It's technical.

**Server API Key Details**

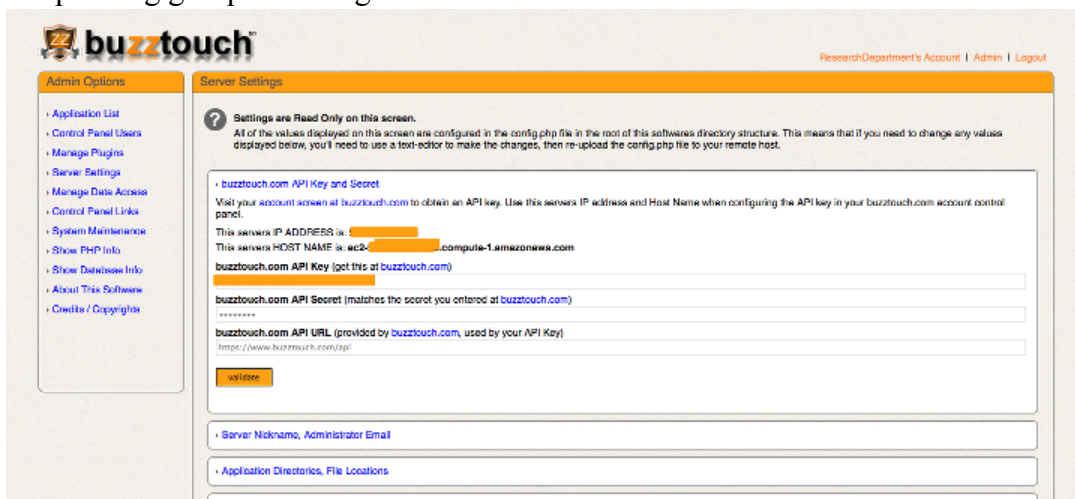
Server Nickname	Research Dept Control Panel
Server IP Address	
Domain Name	42-54-219-242-136.computel-1.amazonaws.com
Control Panel URL	http://wp2-computel-1.amazonaws.com/31-server
API Key	
API URL	http://www.buzztouch.com/wp2
API Calls	7
Maximum Monthly	100
Created	01/11/2014 10:41 PM
Last Request	01/21/2014 02:51 AM
API Key Status	Active

[modify](#) [cancel](#)

- To modify the API key information, click on the “modify” button. A screen on which you can update your API key information will appear.



6. In another tab on your browser, open the settings page for your self-hosted Buzztouch Control Panel. The easiest way is to click on the “Admin” link over near the upper-right corner of an account page on your self-hosted control panel. Then click on the “Server Settings” link at the left side of the screen.
7. On the Server Settings page, click on “buzztouch.com API Key and Secret” to expand the corresponding group of settings.



8. The server's IP ADDRESS should have 54 or something similar as the first number. It should **NOT** have 172 as its first number. If it does, it is because you did not make one of the necessary modifications to `.../BT-server/admin/settings.php` before uploading the Buzztouch files to your EC2 instance. You will copy the IP ADDRESS and HOST NAME (“oranged out” in the image) from here and paste them into the corresponding fields on the “Update API Key” page on the Buzztouch web site that is currently open in another tab in your browser. The labels on the fields on the “Update API Key” page are very helpful, giving you the right format for each field.

- 8.1. The “Server Nickname” can be whatever you want.
- 8.2. Use the IP ADDRESS from the settings page in your Control Panel to fill in the “Server IP Address” field on the buzztouch.com “Update API Key” page. Make sure this is a valid

routable public IP address. It will probably start with 54. It should definitely **NOT** start with 172. Your EC2 instance's public IP address is always available from the information about the instance on the EC2 Management Console page.

- 8.3. Use the HOST NAME from your settings page to fill in the “Server Domain Name” field on the buzztouch.com “Update API Key” page.
- 8.4. Construct a complete URL for the control panel starting with “http://” and your instance's domain and adding “/BT-server” to the end.  
The result should look something like  
`http://ec2-xxx-xxx-xxx-xxx.compute-1.amazonaws.com/BT-server,`  
where, as usual, your EC2 instance's IP address, with “-” instead of “.” as separators, should replace “xxx-xxx-xxx-xxx”  
Use this URL in the “Control Panel URL” field on the buzztouch.com “Update API Key” page.
- 8.5. Enter a password (the same one in both fields) in the “Reset Secret” and “Re-enter Secret” fields on the buzztouch.com “Update API Key” page. Make sure you can remember the password.
- 8.6. Click on the checkbox to accept the Buzztouch terms and condition of use and click on the “save” button.
9. Back on the settings page on your self-hosted Control Panel, enter the API Key and API URL from your API key information page on buzztouch.com in the “buzztouch.com API Key” and “Buzztouch.com API URL” fields. Carefully enter the password (secret) you created for the API key in the “buzztouch.com API Secret” field.

Now the only thing left to do is validate the API key. If you followed these instructions carefully, you should have avoided the most common mistakes.

## **Validating Your API Key**

1. If you've followed these instructions carefully, this should be anti-climactic. In the browser tab with your self-hosted Control Panel's settings open (and specifically, the group), click on the “validate” button. If you see “error saving?” in red, that means something has gone wrong. If you see “**Success! Buzztouch API Key and Secret are valid**” in green text, that means you've done everything right, and you now have your self-hosted Buzztouch Control Panel running on an Amazon EC2 instance.
2. Once everything is really working and your API key is validated, you can delete the `/install` directory on your EC2 instance (this was the third of four steps shown on the final Buzztouch installation page, but I had you hold off doing it until now because you might have needed something in the `/install` directory to get your Control Panel working). Execute the following command in the Terminal in which you're connected to your EC2 instance (you should be in directory `/var/www/BT-server`) and are the superuser:  

```
rm -rf install
```

## **What To Do After It's Working**

1. Once you've gotten your self-hosted Control Panel working, I suggest making an AMI (Amazon Machine Image) of your server. Do this by selecting the checkbox next to your instance on your EC2 Management Console “Instances” page, clicking on the “Actions ∨” button and selecting “Create Image” from the menu that appears.
2. You should occasionally make a backup using the “create image” functionality as described in the step above. Additionally, you should occasionally make sure all the software on your instance is up to date by logging in to your instance through a Terminal (using your key pair as described several times in this document) and executing the following commands:

```
sudo -s
apt-get update
apt-get dist-upgrade
```
3. Now that your control panel is working, use it to do things like making sure all the plugins you have installed at buzztouch.com are also installed on your self-hosted Control Panel. You'll then be ready to start making apps!

## **Acknowledgements**

As mentioned throughout the text of this “how to” guide, I was only able to get my self-hosted Control Panel working because of the help of Buzztouch Forum veterans like SmugWimp, Niraj Shah, and PSMDanny. Additionally, I was able to spare the veterans a bunch of potentially annoying questions whose answers would basically be polite variations on “RTFM” by reading the documents already available from the buzztouch web site.

The three documents I used most were:

- The existing “how to” guide on installing the Buzztouch Control Panel on an EC2 instance. The title of this document is “Buzztouch Server 2.0 with Amazon EC2,” and the file name under which I found it is <https://www.buzztouch.com/files/howtos/Amazon-EC2-with-buzztouch.pdf>
- The official Buzztouch instructions for setting up a self-hosted Control Panel, which I found at <https://www.buzztouch.com/files/howtos/bt-self-hosted-3.0.0.pdf>
- The alternate step-by-step tutorial on setting up a self-hosted Buzztouch Control Panel, which I found at <https://www.buzztouch.com/files/howtos/BuzzTouchServerTutorial.pdf>