The required Apache configuration and commands can vary quite a bit based on the platform you're using. If you use Ubuntu Server as recommended, the preceding should all work smoothly. If not, you may need to read up a bit on how to configure Apache for your specific platform.

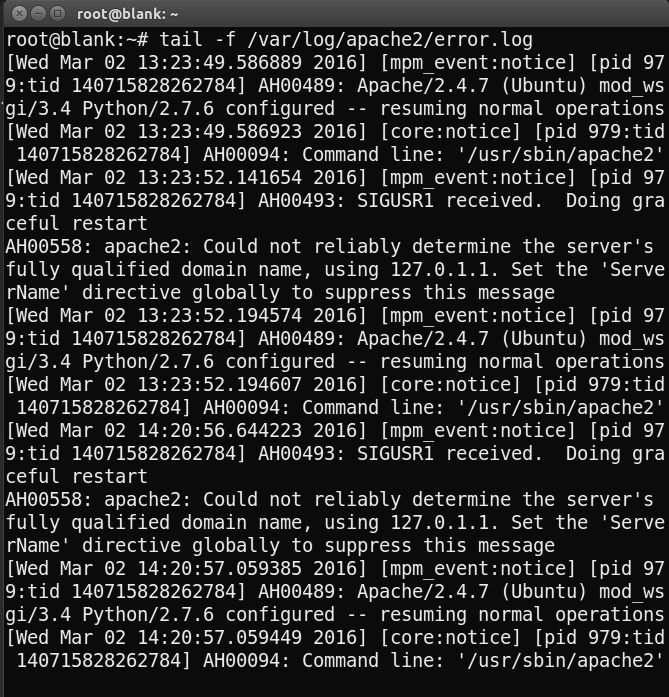
You should note reloading web server apache2 in the output. If errors are displayed, then you probably misconfigured something in the preceding command. If this is the case, read the error message carefully and go back over the previous steps to take a look at why things didn't work as expected.

To test that everything is working, open a web browser on your local machine and type your IP address into the address bar again. You should see **Hello, World!** displayed in your browser instead of the default Apache page that we saw before.

If you get **Error 500** instead, it means that our application fell over for some reason. Fear not; it's better that you get used to dealing with this error now, when the fix will probably be simple, than later on, when we've added more components that could break or be misconfigured. To find out what went wrong, run the following command on your VPS:

**sudo tail –f /var/log/apache2/error.log**

The tail command simply outputs the last several lines of the file passed as an argument. The -f is for follow, which means that the output will be updated if the file changes. If you can't immediately work out which lines are indicative of the error we're looking for, visit the site in your web browser on your local machine again, and you'll see the output from the tail command be updated accordingly. The following screenshot shows the output from the tail command when there are no errors; however, if anything goes wrong, you'll see the error output printed among all the info messages.



Some possible tripping points are incorrectly configured WSGI and Apache files (make sure that your WSGIDaemonProcess and daemon name match, for example) or incorrectly configured Python (you may forget to install Flask on your VPS). If you can't figure out what the error message means, an Internet search for the message (removing the error-specific parts of your app, such as names and paths) will usually point you in the right direction. Failing this, there are strong and very friendly Flask and WSGI communities on Stack Overflow and Google Groups, and there's normally someone willing to help beginners. Remember that if you're having a problem and can't find an existing solution online, don't feel bad for asking; you'll help countless people facing issues similar to yours.