Introduction

This document is meant to show a quick tutorial to streamlining your ssh connections. By the end of this document you will have enabled the ability to enter commands of this flavor:

`ssh` *my-remote-server*

`scp` *my-remote-server* `local_file` : `remote_destination`

without needing to enter your password. This is a method called **ssh aliasing**. Anyway, without further adieu, let's set it up.

Setup

1. Enter your `/.ssh` folder.
2. Create a file called `config` and enter the following information (verbatim):

   ```
   Host *
   AddKeysToAgent yes
   UseKeychain yes
   IdentityFile  /.ssh/id_rsa
   ```

   In the same file, enter the your custom remote host information:

   `Host` *custom-host-alias*
   `User` *your-username-on-this-host*
   `Hostname` `the_actual_hostname`

   In the case of WePanic, the only custom entry in this part of the file is the `Host` parameter. The Username is `bloodletter` and the Hostname is `wepanic-dl.eastus.cloudapp.azure.com`.
   You have now aliased your ssh. Trying `ssh` *custom-host-alias* will work, but still prompt you with a password. We'll fix that in the next step.
3. Make an ssh key pair:

   ```
   ssh-keygen -t rsa
   ```

   Press `enter` through the prompted messages; you DO NOT want an alternate location nor a passphrase.
4. Install the ssh key on the remote machine.

   ```
   cat  /.ssh/id_rsa.pub | ssh user@remote "mkdir -p  /.ssh; cat >>  /.ssh/authorized_keys"
   ```

   Make sure to use `>>` not `>` to *append* to the file rather than overwriting it!