## Install Apache on the Load Balancer Server

Log into SSH. Afterwards, install Apache web server on the load balancer server from your Linux distribution (distro) repositories (e.g. CentOS, Ubuntu, etc.).

For better server security, enable the ModSecurity Apache module as well.

## Enable Proxy Server Modules

You’ll need the following Apache modules for load balancing:

* proxy
* proxy\_http
* Proxy\_balancer (for Balancer Manager)
* lbmethod\_bytraffic (for round robin load balancing)

Ensure the necessary modules are enabled.

CentOS:

Copy

httpd -M | grep 'proxy\|lbmethod'

Debian:

Copy

apache2ctl -M | grep 'proxy\|lbmethod'

If any module listed above is not installed or enabled, enable it.

CentOS:

Copy

yum install [module]

Debian:

Copy

a2enmod [module]

Restart Apache if you had to enable a module.

CentOS:

Copy

systemctl restart httpd

Debian:

Copy

Systemctl restart apache2

## Configure Apache Load Balancing

Create and edit a new load balancer file using Nano, Vim, Emacs, etc.

CentOS:

Copy

nano /etc/httpd/conf.d/loadbalancer.conf

Debian:

Copy

Nano /etc/apache2/conf-enabled/loadbalancer.conf

The name of the file doesn’t matter. What matters is that the file is titled for you to easily remember what it is, is in the correct directory for your distro, and ends with the .conf file extension.

Add the code block below, but replace the following:

* serverpool with your preferred name for the balancer function
* IP addresses with the IP, domain, or server hostname of your web servers along with the port number
* sitePass with the URI which will redirect to the server pool members

Copy

<proxy balancer://serverpool>

        BalancerMember http://1.2.3.4:80

        BalancerMember http://5.6.7.8:80

        ProxySet lbmethod=bytraffic

</proxy>

ProxyPass "/site" "balancer://serverpool/"

ProxyPassReverse "/site" "balancer://serverpool/"

If you plan to add domains with a valid SSL certificate, you may need to install the mod\_ssl Apache module as well.

Save changes and restart Apache: systemctl restart httpd or systemctl restart apache2

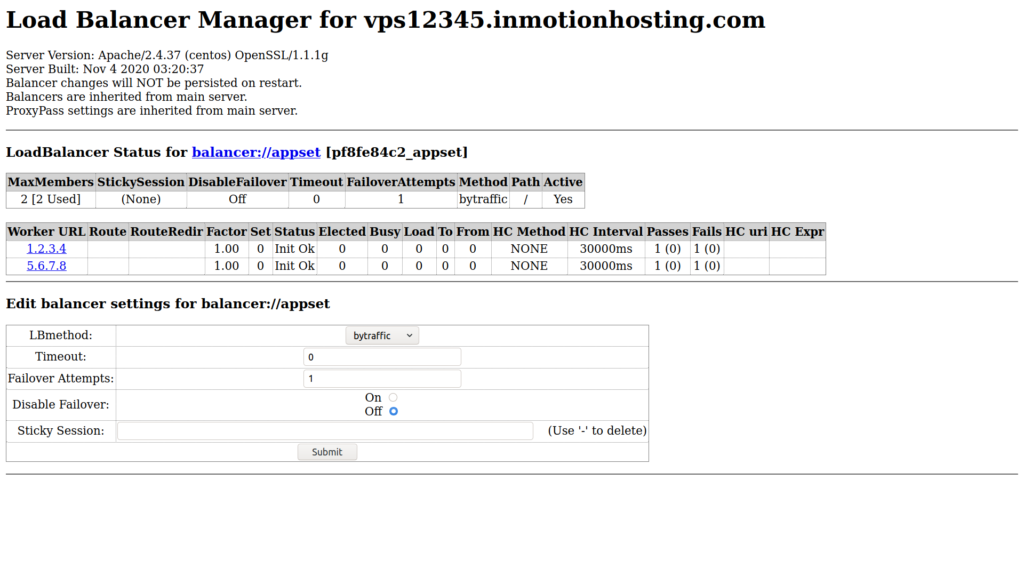
Terminal tip: Use your up arrow key to quickly reuse commands from your .bash\_history file.

Open the URL to your load balancer function in a web browser. It should display the website from one of the balancer members. When you refresh the page, it should show the next balancer member. If not, try a private browsing session.

If your load balancer doesn’t work, start troubleshooting with your Apache error\_log.

## Configure Apache Load Balancer Manager

The Apache Balancer Manager allows you to monitor and manage your load balancing configuration. This is optional but a light-weight monitoring solution. For more advanced options, look into ****security information and event management (SIEM)**** solutions such as [ELK Stack](https://www.inmotionhosting.com/support/edu/software/how-to-install-elasticsearch/) or [Splunk](https://www.inmotionhosting.com/support/security/install-splunk/). Ensure you have proxy\_balancer\_module enabled before continuing below.

Apache Load Balancer Manager settings

To get started, create and edit a new Apache configuration file.

CentOS:

Copy

nano /etc/httpd/conf.d/loadbalancer-manager.conf

Debian:

Copy

nano /etc/apache2/conf-enabled/loadbalancer-manager.conf

Again, the name of the file doesn’t matter, only the directory and .conf file extension.

Add the following code block to the file, replacing “lb-manager” with your preferred URI for accessing the page.

Copy

<location "/lb-manager">

        SetHandler balancer-manager

        allow from all

</location>

By default, the page is publicly accessible. To configure Apache Load Balancer Manager securely, add the next code block above the </location> line. Then, use htpasswd to create user credentials to restrict the page to authorized users only. Change “AuthName” to match your location URI and “AuthUserFile” with the appropriate file path per your distro.

CentOS:

Copy

AuthType "basic"

AuthName "lb-manager"

AuthUserFile /etc/httpd/htpasswd

Require valid-user

Debian:

Copy

AuthType "basic"

AuthName "lb-manager"

AuthUserFile /etc/apache2/htpasswd

Require valid-user

Create a htpasswd file. Replace “admin” at the end with an obscure username.

CentOS:

Copy

htpasswd -c /etc/httpd/htpasswd admin

Debian:

Copy

htpasswd -c /etc/apache2/htpasswd admin

You’ll be prompted for a password. Enter the password twice. Save it in a password manager like KeePass.

Add the following to the bottom of your load balancer configuration file, replacing “lb-manager” to match your balancer manager location:

Copy

ProxyPass "/lb-manager" "!"

Restart Apache.

Open the URL in your web browser to access your balancer-manager:  
http://[serverhostname]/[lb-manager]

If you used htpasswd, you’ll be prompted for an username and password before you can continue.

Load Balancer Manager will display the following notable information:

* Web server version
* Whether Load Balancer changes are retained upon system reboot
* LoadBalancer members, or work URLs, along with server load statistics

You can make changes by clicking the balancer://[serverpoolname] link. Options to edit balancer settings include:

* LBMethod – bytraffic, heartbeat (ready capacity over time but not fully idle), bybusyness, byrequests (you’ll need to have the appropriate modules enabled for the respective method)
* Timeout
* Failover Attempts
* Disable Failover
* Sticky Session to ensure users are bound to a specific server

By default, changes here don’t persist on restart. To change this, add the line below to the bottom of your load balancer configuration file.

Copy

BalancerPersist On

Restart Apache. Upon refreshing Apache Load Balancer Manager, you’ll see: “Balancer changes will be persisted on restart.”