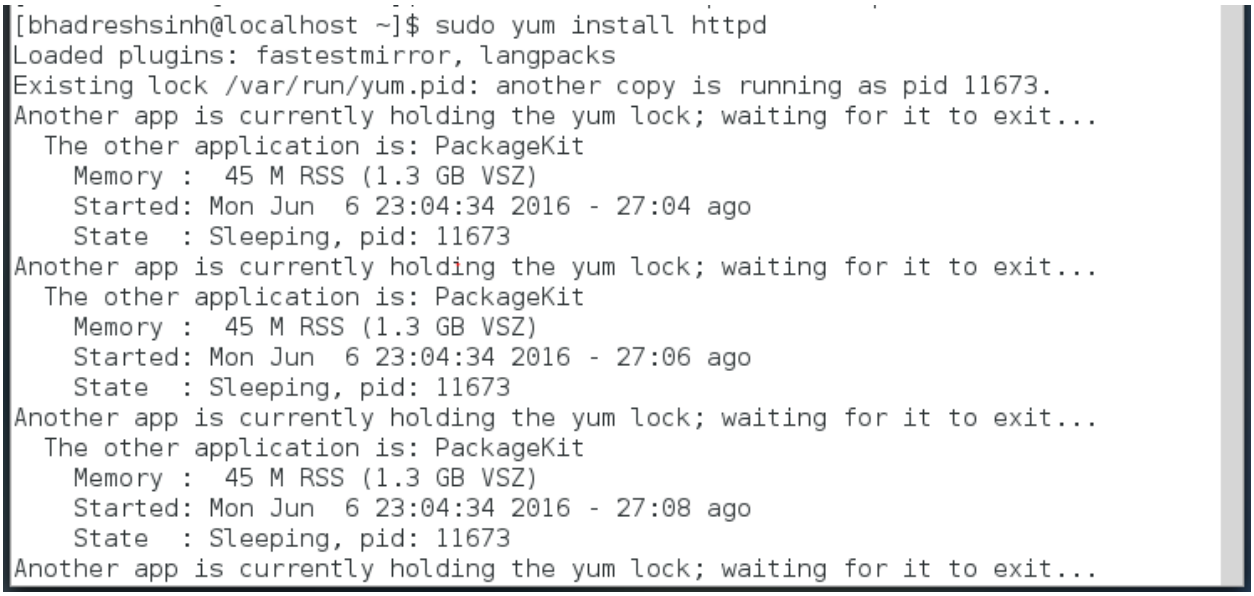
How to optimize Apache performance on Centos-7

Step One — Install Apache

The Apache web server is currently the most popular web server in the world, which makes it a great default choice for hosting a website.

We can install Apache easily using CentOS's package manager, yum. A package manager allows us to install most software pain-free from a repository maintained by CentOS. You can learn more about how to use yum here.

For our purposes, we can get started by typing these commands:



MaxKeepAliveRequests

MaxKeepAliveRequests is the maximum number of requests to serve on a TCP connection. It limits the number of requests allowed per connection. If it is set to 0, unlimited requests will be allowed. You can set it to any value you desire.

Keep this setting to a high value for maximum server performance. The recommended value of MaxKeepAliveRequests is 500.

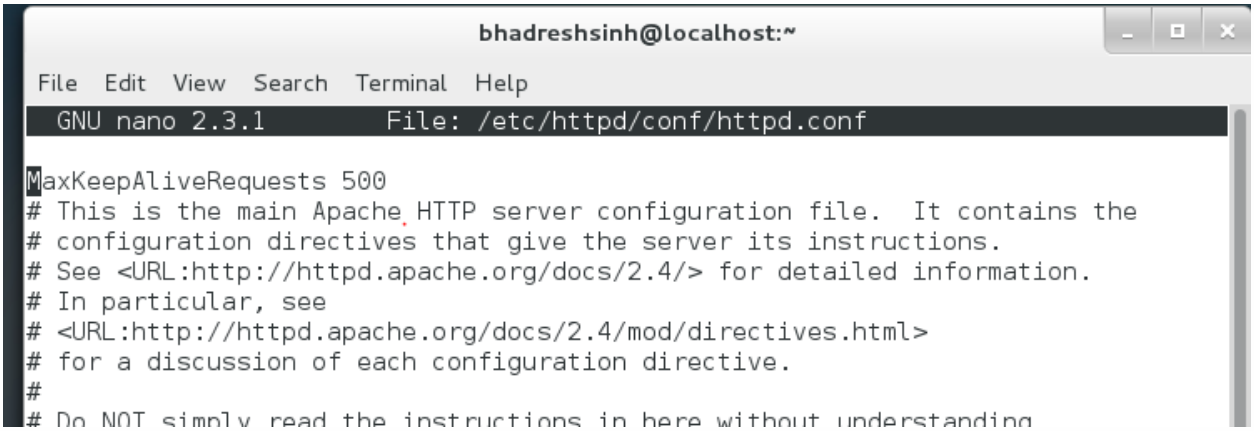
To change this setting, edit the Apache configuration file:

```
sudo nano /etc/httpd/conf/httpd.conf
```

Add the following line:

```
MaxKeepAliveRequests 500
```

Save and close the file when you are finished.



KeepAliveTimeout

KeepAliveTimeout defines the number of seconds Apache will wait for the new request from connected clients before closing the connection. (Once the server receives a request, the Timeout directive applies instead.)

By default Keepalive is disabled in CentOS 7. If Keepalive is set to on, it is a good idea to set the KeepAliveTimeout value low.

The recommended KeepAliveTimeout can be between 1 to 5.

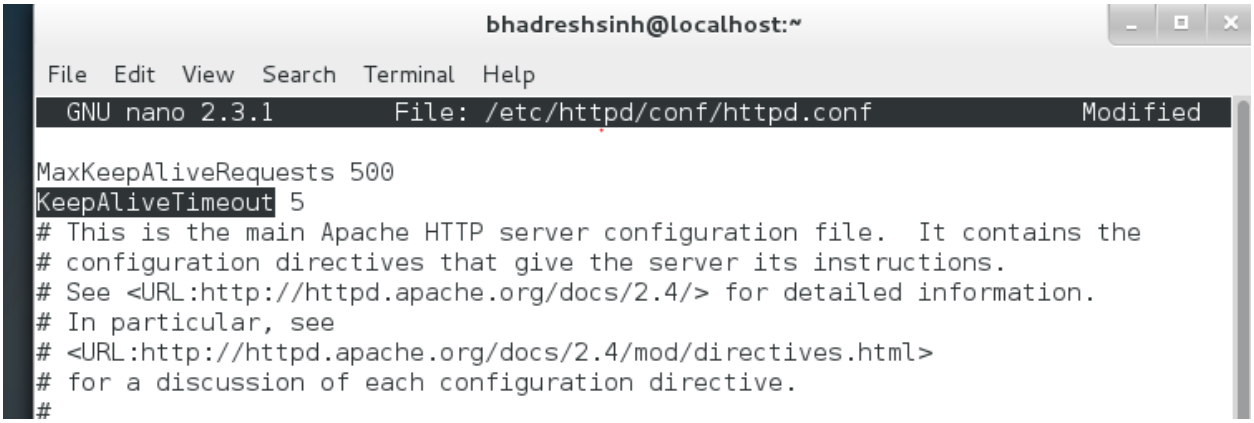
You can do this by editing Apache configuration file:

```
sudo nano /etc/httpd/conf/httpd.conf
```

Add the following line:

```
KeepAliveTimeout 5
```

Save and close the file when you are finished.



KeepAlive

KeepAlive sets whether the server allows more than one request per connection. It can be used to prevent any one client from consuming too much of the server's resources.

By default KeepAlive is disabled in CentOS 7. Once the Apache server is getting requests from hundreds and thousands of IPs at once, this setting should be On.

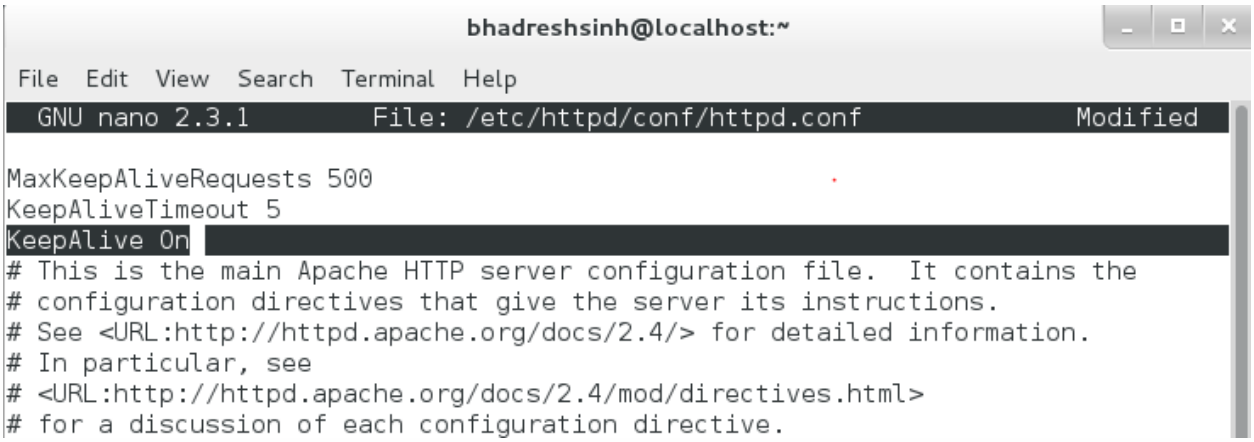
You can enable this setting by editing Apache configuration file:

```
sudo nano /etc/httpd/conf/httpd.conf
```

Add the following line:

```
KeepAlive On
```

Save and close the file when you are finished.



Configure MPM Prefork

One reason for poor Apache performance is that Apache is having trouble coping with the load. The Apache MPM (Multi-Processing Module) can help.

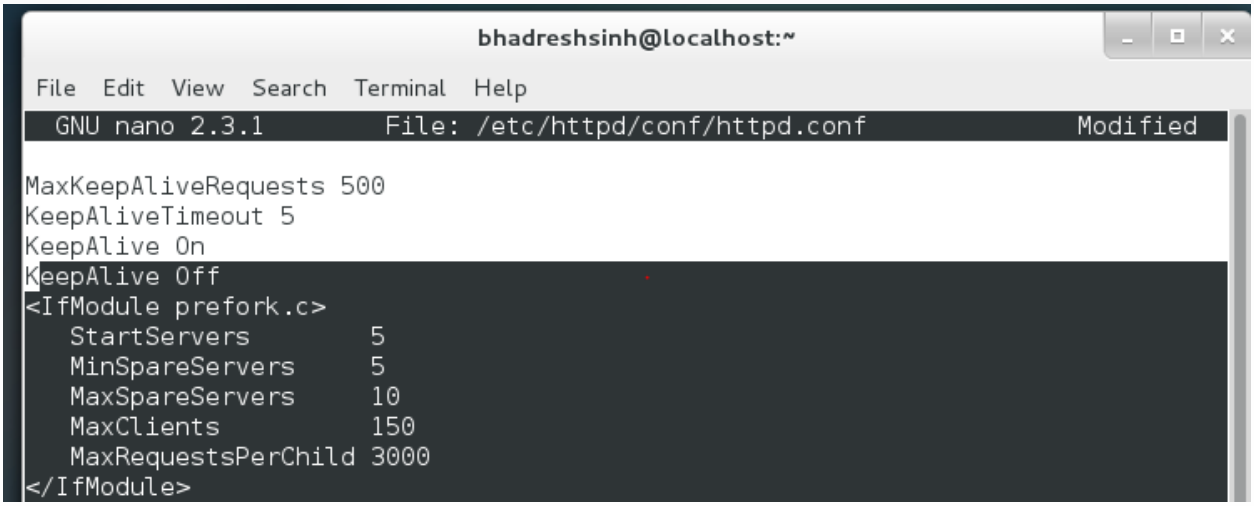
mpm_prefork_module is included and enabled in the default Apache installation on CentOS 7. To confirm this run the following command:

```
sudo apachectl -t -D DUMP_MODULES |grep mpm
```

You should see mpm_prefork_module (shared) if mod_deflate is installed and enabled.

You can make Apache performance better using the Apache MPM prefork module. To do this, set the following parameters in your Apache configuration file:

```
sudo nano /etc/httpd/conf/httpd.conf
```



DNS Lookups

The biggest reason for Apache web server slowdowns is the time required to perform DNS lookups. Apache will record the full host name of each incoming client connection in its access.log file. Resolving each one eats up a significant chunk of time.

The HostnameLookups option enables DNS lookup so that hostnames can be logged instead of the IP address. By defaultHostnameLookups is Off in Apache.

You can verify that this is the case by editing the Apache config file:

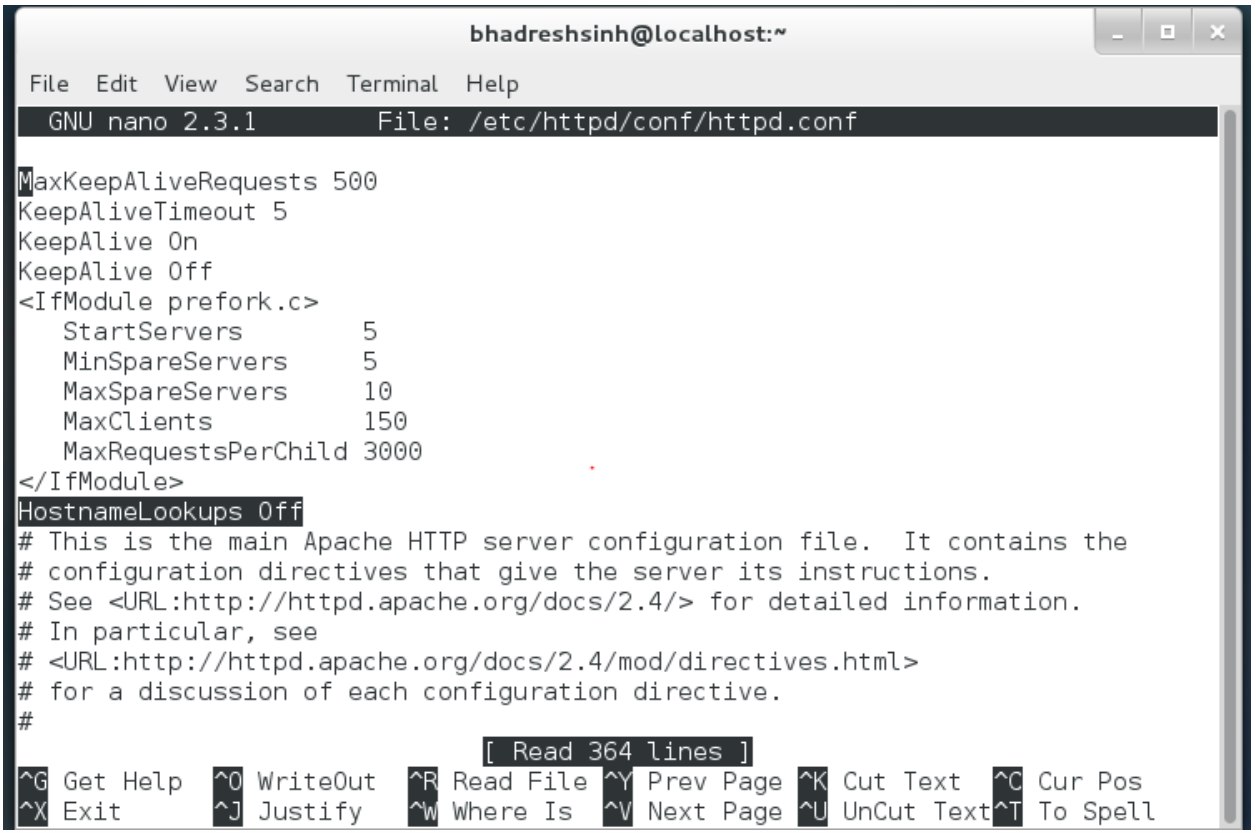
```
sudo nano /etc/httpd/conf/httpd.conf
```

Be sure the HostnameLookups line reads:

```
HostnameLookups Off
```

Save and close the file when you are finished, then restart Apache to reflect changes.

```
sudo apachectl restart
```




```
[bhadreshsinh@localhost ~]$ sudo nano /etc/httpd/conf/httpd.conf
[bhadreshsinh@localhost ~]$ sudo nano /etc/httpd/conf/httpd.conf
[bhadreshsinh@localhost ~]$ sudo apachectl restart
[bhadreshsinh@localhost ~]$ sudo nano /etc/httpd/conf/httpd.conf
[bhadreshsinh@localhost ~]$ sudo apachectl restart
[bhadreshsinh@localhost ~]$ sudo nano /etc/httpd/conf/httpd.conf
[bhadreshsinh@localhost ~]$
```

Always keep Apache updated to its latest version

```
[bhadreshsinh@localhost ~]$ httpd -v
Server version: Apache/2.4.6 (CentOS)
Server built:   May 12 2016 10:27:23
[bhadreshsinh@localhost ~]$
```

If you are using a Kernel older than 2.4, consider upgrading now

```
Complete!
[bhadreshsinh@localhost ~]$ httpd -v
Server version: Apache/2.4.6 (CentOS)
Server built:   May 12 2016 10:27:23
[bhadreshsinh@localhost ~]$ uname -r
3.10.0-123.el7.x86_64
[bhadreshsinh@localhost ~]$
```

Choose the Multi-Processing Module (MPM) that works best for your case

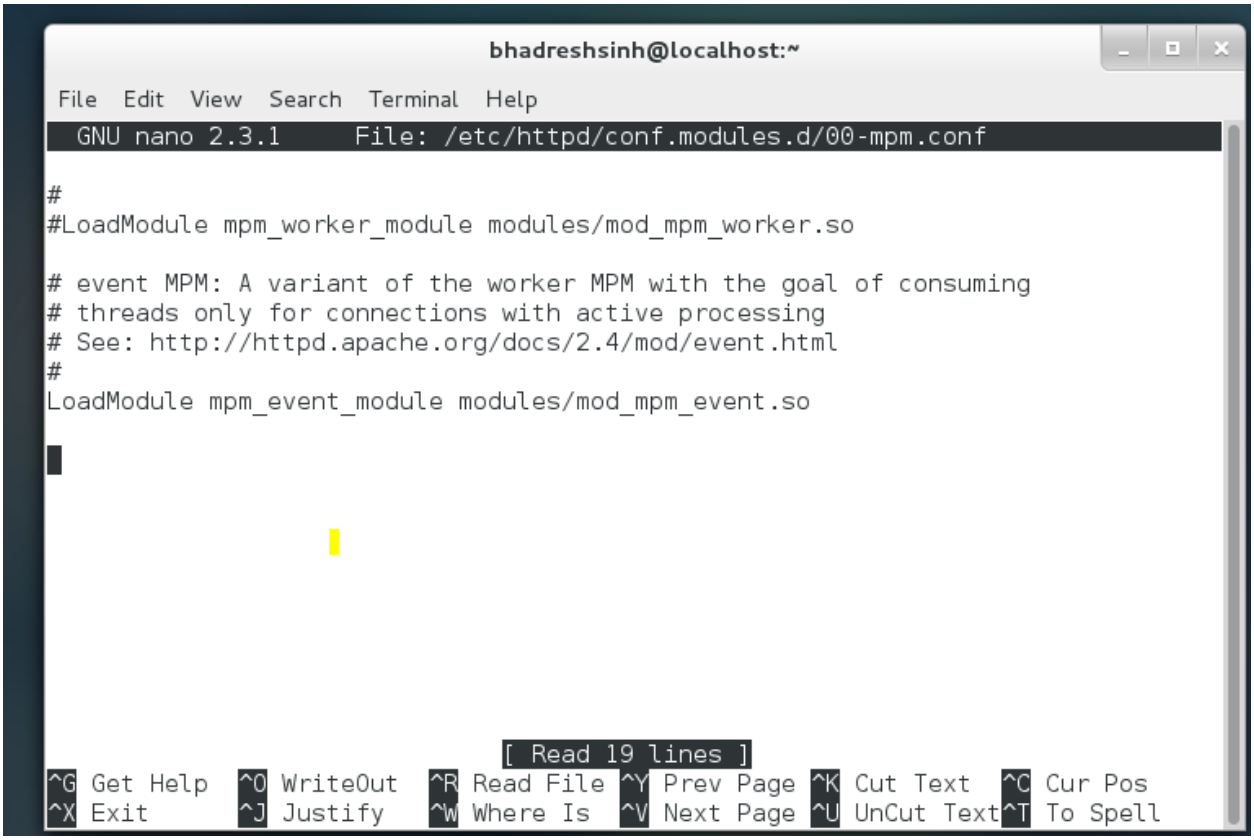
```
[bhadreshsinh@localhost ~]$ httpd -V
AH00558: httpd: Could not reliably determine the server's fully qualified domain
 name, using localhost.localdomain. Set the 'ServerName' directive globally to s
uppress this message
Server version: Apache/2.4.6 (CentOS)
Server built:   May 12 2016 10:27:23
Server's Module Magic Number: 20120211:24
Server loaded:  APR 1.4.8, APR-UTIL 1.5.2
Compiled using: APR 1.4.8, APR-UTIL 1.5.2
Architecture:   64-bit
Server MPM:     prefork
  threaded:     no
    forked:     yes (variable process count)
Server compiled with....
 -D APR_HAS_SENDFILE
 -D APR_HAS_MMAP
```

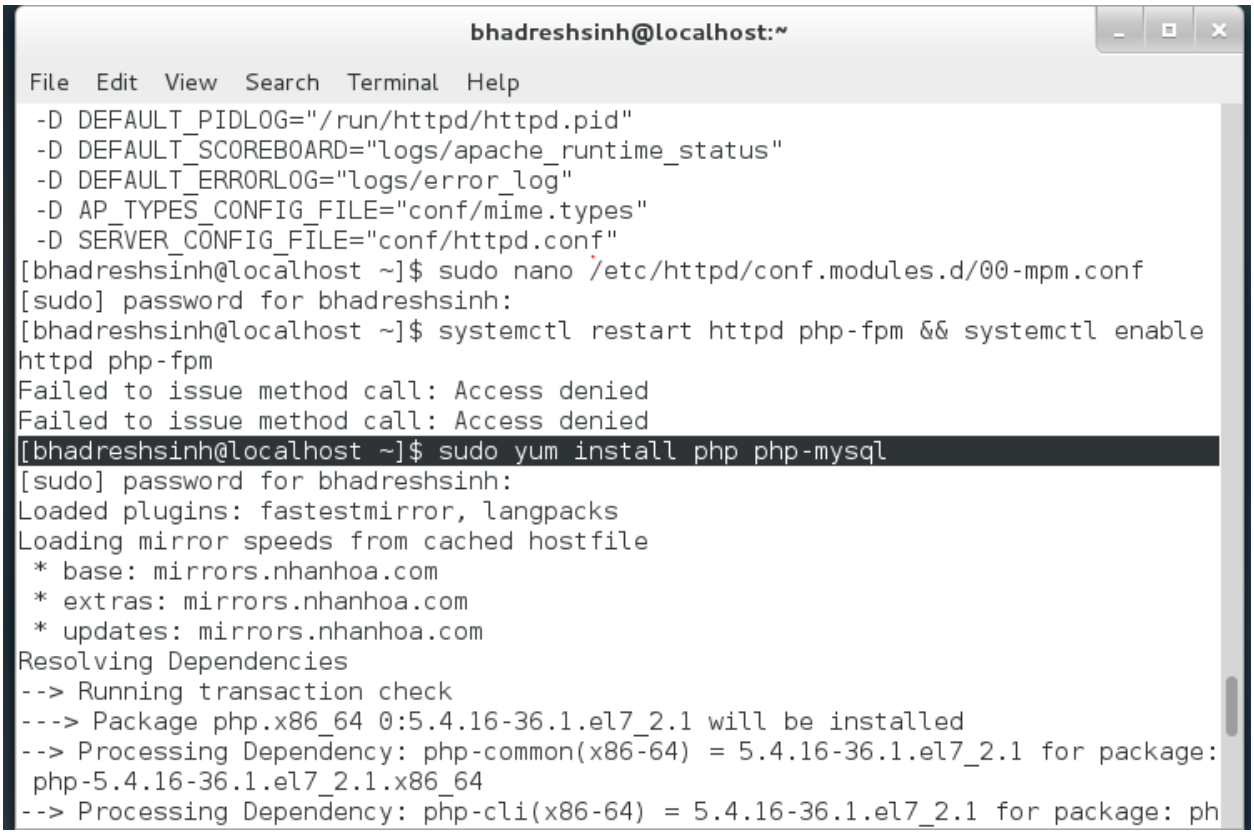
To change this, you will need to edit:

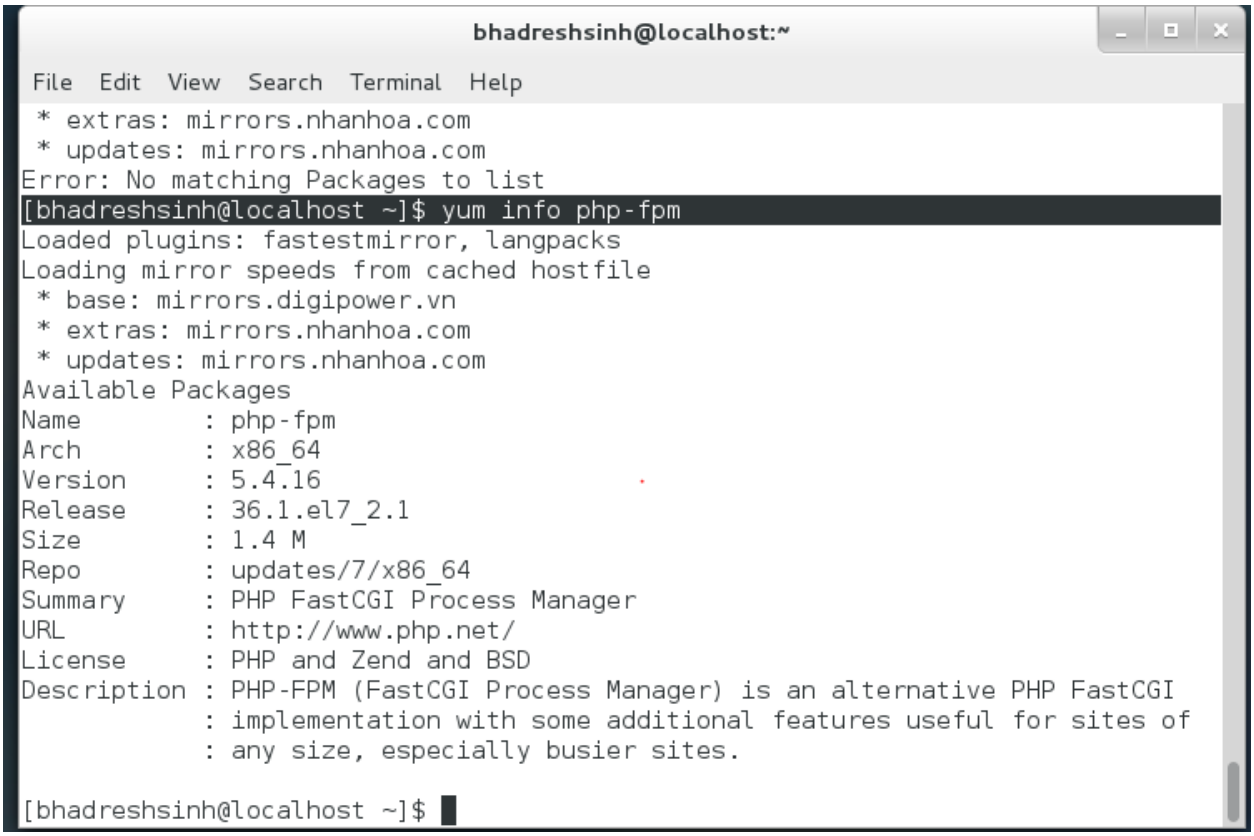
```
# /etc/httpd/conf.modules.d/00-mpm.conf
```

Where <mpm> can be **mpm_event**, **mpm_worker**, or **mpm_prefork**.
and uncomment the line that loads the desired module like so:

```
LoadModule mpm_event_module modules/mod_mpm_event.so
```


bhadreshsinh@localhost:~
File Edit View Search Terminal Help
GNU nano 2.3.1 File: /etc/httpd/conf.modules.d/00-mpm.conf
#
#LoadModule mpm_worker_module modules/mod_mpm_worker.so
event MPM: A variant of the worker MPM with the goal of consuming
threads only for connections with active processing
See: http://httpd.apache.org/docs/2.4/mod/event.html
#
LoadModule mpm_event_module modules/mod_mpm_event.so
[Read 19 lines]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell


bhadreshsinh@localhost:~
File Edit View Search Terminal Help
-D DEFAULT_PIDLOG="/run/httpd/httpd.pid"
-D DEFAULT_SCOREBOARD="logs/apache_runtime_status"
-D DEFAULT_ERRORLOG="logs/error_log"
-D AP_TYPES_CONFIG_FILE="conf/mime.types"
-D SERVER_CONFIG_FILE="conf/httpd.conf"
[bhadreshsinh@localhost ~]$ sudo nano /etc/httpd/conf.modules.d/00-mpm.conf
[sudo] password for bhadreshsinh:
[bhadreshsinh@localhost ~]$ systemctl restart httpd php-fpm && systemctl enable
httpd php-fpm
Failed to issue method call: Access denied
Failed to issue method call: Access denied
[bhadreshsinh@localhost ~]$ sudo yum install php php-mysql
[sudo] password for bhadreshsinh:
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
* base: mirrors.nhanhoa.com
* extras: mirrors.nhanhoa.com
* updates: mirrors.nhanhoa.com
Resolving Dependencies
--> Running transaction check
---> Package php.x86_64 0:5.4.16-36.1.el7_2.1 will be installed
--> Processing Dependency: php-common(x86-64) = 5.4.16-36.1.el7_2.1 for package:
php-5.4.16-36.1.el7_2.1.x86_64
--> Processing Dependency: php-cli(x86-64) = 5.4.16-36.1.el7_2.1 for package: ph


bhadreshsinh@localhost:~
File Edit View Search Terminal Help
* extras: mirrors.nhanhoa.com
* updates: mirrors.nhanhoa.com
Error: No matching Packages to list
[bhadreshsinh@localhost ~]$ yum info php-fpm
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
* base: mirrors.digipower.vn
* extras: mirrors.nhanhoa.com
* updates: mirrors.nhanhoa.com
Available Packages
Name : php-fpm
Arch : x86_64
Version : 5.4.16
Release : 36.1.el7_2.1
Size : 1.4 M
Repo : updates/7/x86_64
Summary : PHP FastCGI Process Manager
URL : http://www.php.net/
License : PHP and Zend and BSD
Description : PHP-FPM (FastCGI Process Manager) is an alternative PHP FastCGI
: implementation with some additional features useful for sites of
: any size, especially busier sites.
[bhadreshsinh@localhost ~]$