Penetration Testing Lab Setup: Memcached

February 16, 2019 By Raj Chandel

In this article, we are going to learn about pen-testing in Memcached lab setup in Ubuntu 18.04. Memcached server is used by corporations in order to increase the speed of their network as it helps to store frequently used data. This helps to take the load of the hardware and decrease the time taken.

Table of Contents

Introduction to Memcached

Memcached is a distributed memory object caching system. It's an open source and without any cost tool. It is used to speed up web applications by using a database from the cache memory. It is an in-memory key-value store for little bits of self-assertive information (strings, objects) that is extracted from database calls, API calls, or page rendering. Memcached is basic however capable of advancing speed arrangement, ease of advancement, and understands numerous issues confronting expansive information caches. Its API is accessible for most prevalent languages.

Memcached Installation

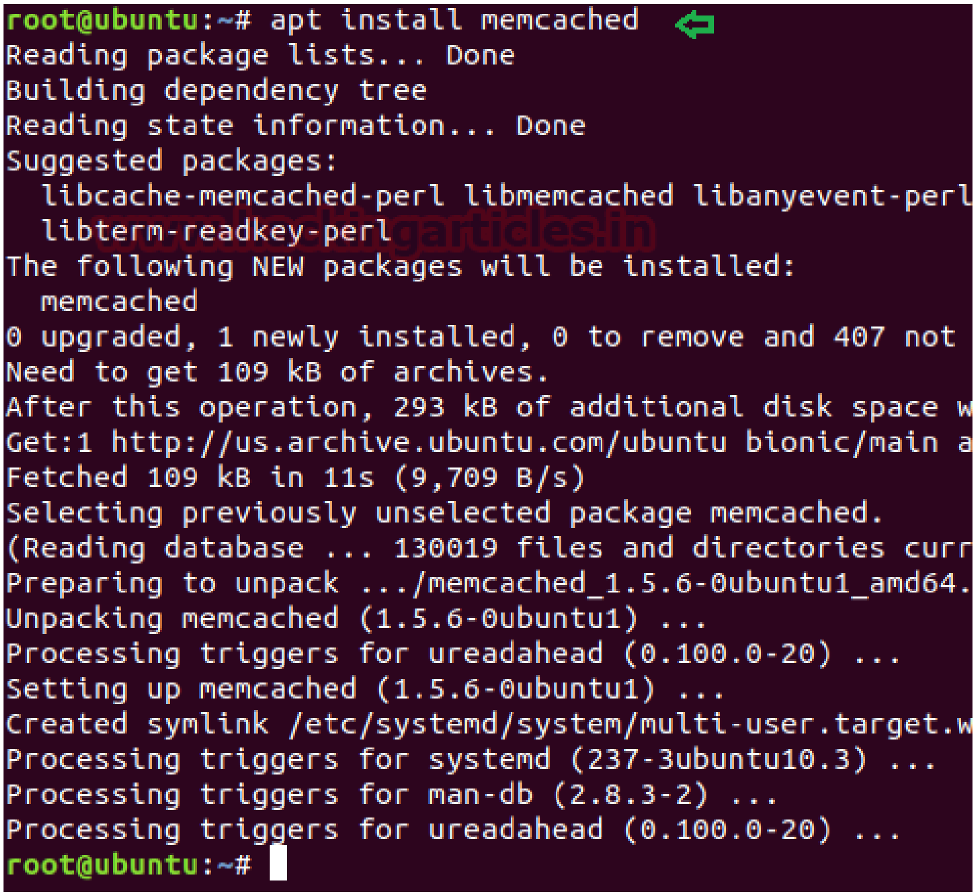
To install, boot up your Ubuntu machine and open the terminal.

Note: Apache2 should be installed before installing Memcached. You can easily install Apache2 by just typing in a simple command.

```
apt install apache2
```

Now that we are all done, let's setup Memcached by typing the commands shown below.

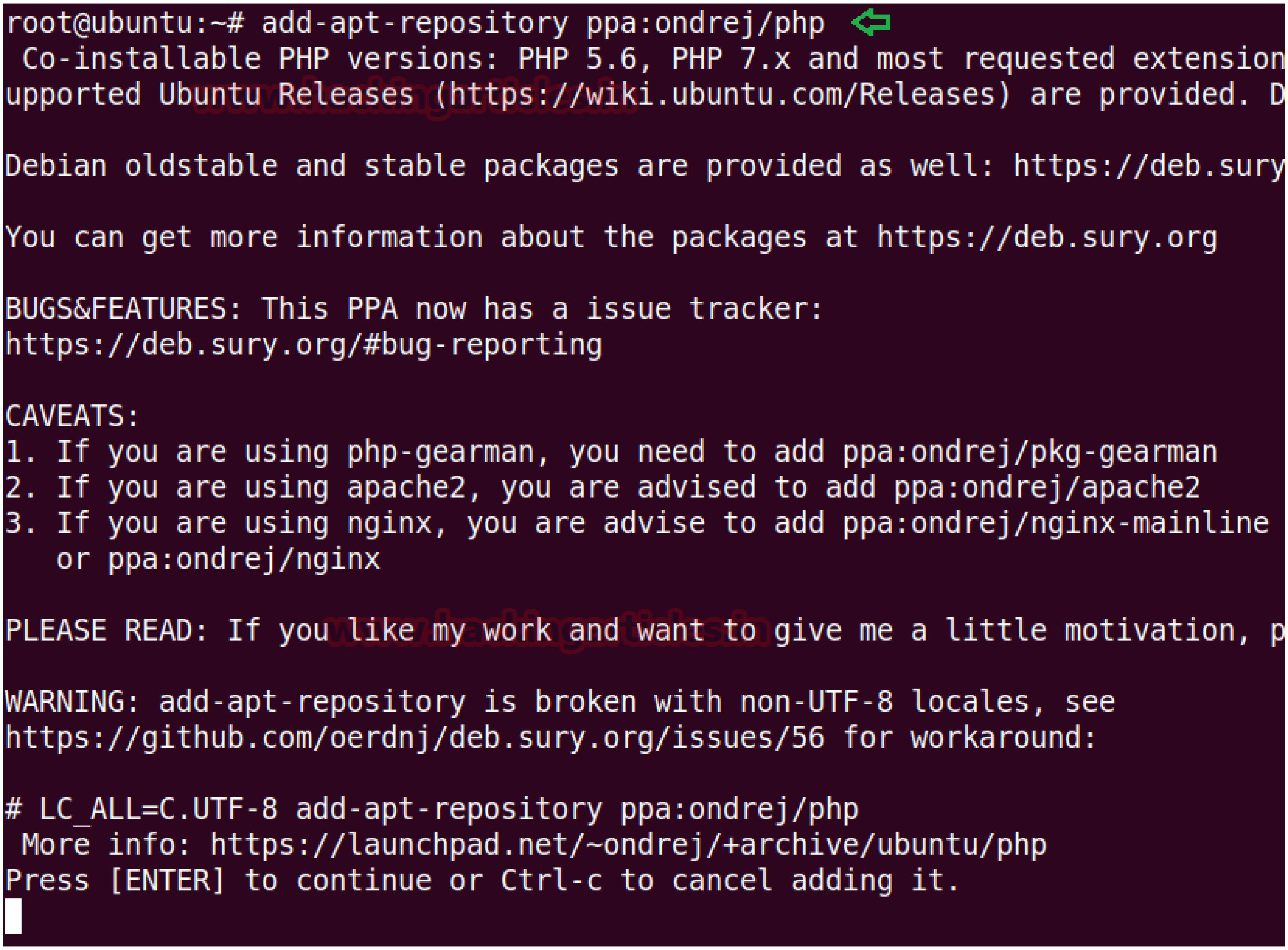
```
apt install memcached
```



```
root@ubuntu:~# apt install memcached
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  libcache-memcached-perl libmemcached libanyevent-perl
  libterm-readkey-perl
The following NEW packages will be installed:
  memcached
0 upgraded, 1 newly installed, 0 to remove and 407 not
Need to get 109 kB of archives.
After this operation, 293 kB of additional disk space w
Get:1 http://us.archive.ubuntu.com/ubuntu bionic/main a
Fetched 109 kB in 11s (9,709 B/s)
Selecting previously unselected package memcached.
(Reading database ... 130019 files and directories curr
Preparing to unpack .../memcached_1.5.6-0ubuntu1_amd64.
Unpacking memcached (1.5.6-0ubuntu1) ...
Processing triggers for ureadahead (0.100.0-20) ...
Setting up memcached (1.5.6-0ubuntu1) ...
Created symlink /etc/systemd/system/multi-user.target.w
Processing triggers for systemd (237-3ubuntu10.3) ...
Processing triggers for man-db (2.8.3-2) ...
Processing triggers for ureadahead (0.100.0-20) ...
root@ubuntu:~#
```

After installing Memcached add ppa:ondrej/php PPA in your Ubuntu system's repository to download and install the latest version of PHP available. Follow the commands as shown below.

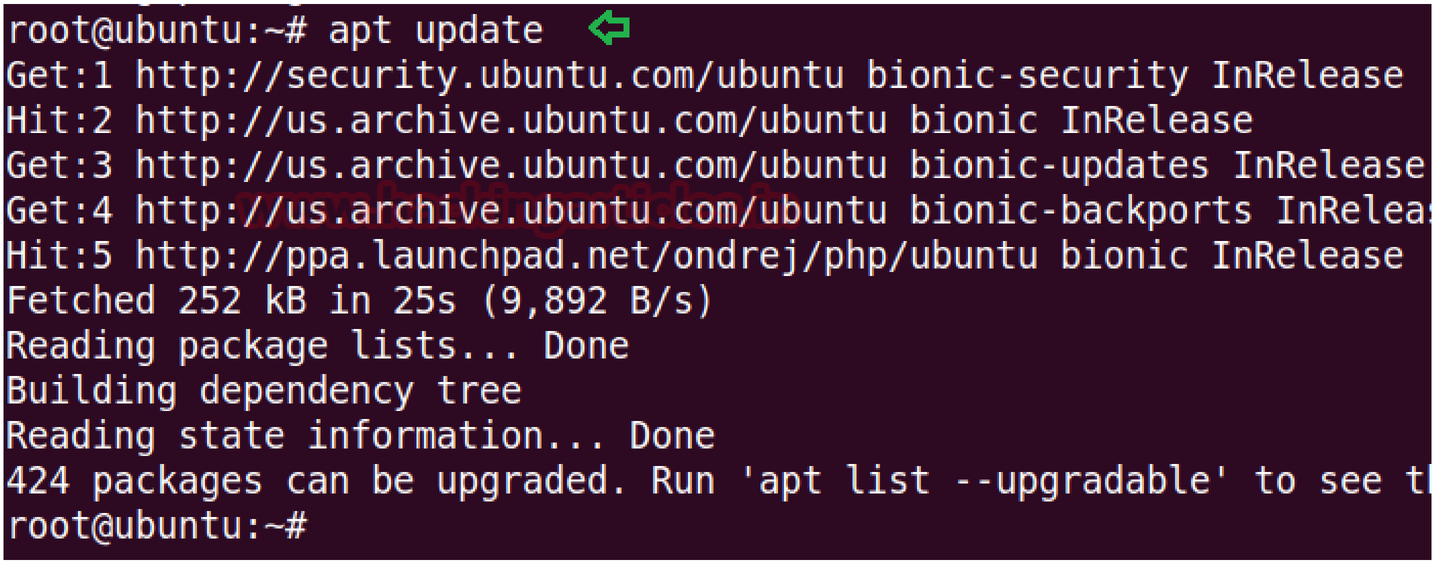
```
add-apt-repository ppa:ondrej/php
```



```
root@ubuntu:~# add-apt-repository ppa:ondrej/php
 Co-installable PHP versions: PHP 5.6, PHP 7.x and most requested extension
upported Ubuntu Releases (https://wiki.ubuntu.com/Releases) are provided. D

Debian oldstable and stable packages are provided as well: https://deb.sury

You can get more information about the packages at https://deb.sury.org

BUGS&FEATURES: This PPA now has a issue tracker:
https://deb.sury.org/#bug-reporting

CAVEATS:
1. If you are using php-gearman, you need to add ppa:ondrej/pkg-gearman
2. If you are using apache2, you are advised to add ppa:ondrej/apache2
3. If you are using nginx, you are advise to add ppa:ondrej/nginx-mainline
   or ppa:ondrej/nginx

PLEASE READ: If you like my work and want to give me a little motivation, p

WARNING: add-apt-repository is broken with non-UTF-8 locales, see
https://github.com/oerdnj/deb.sury.org/issues/56 for workaround:

# LC_ALL=C.UTF-8 add-apt-repository ppa:ondrej/php
 More info: https://launchpad.net/~ondrej/+archive/ubuntu/php
Press [ENTER] to continue or Ctrl-c to cancel adding it.
```

After adding the repository, update the system by typing in the following command.

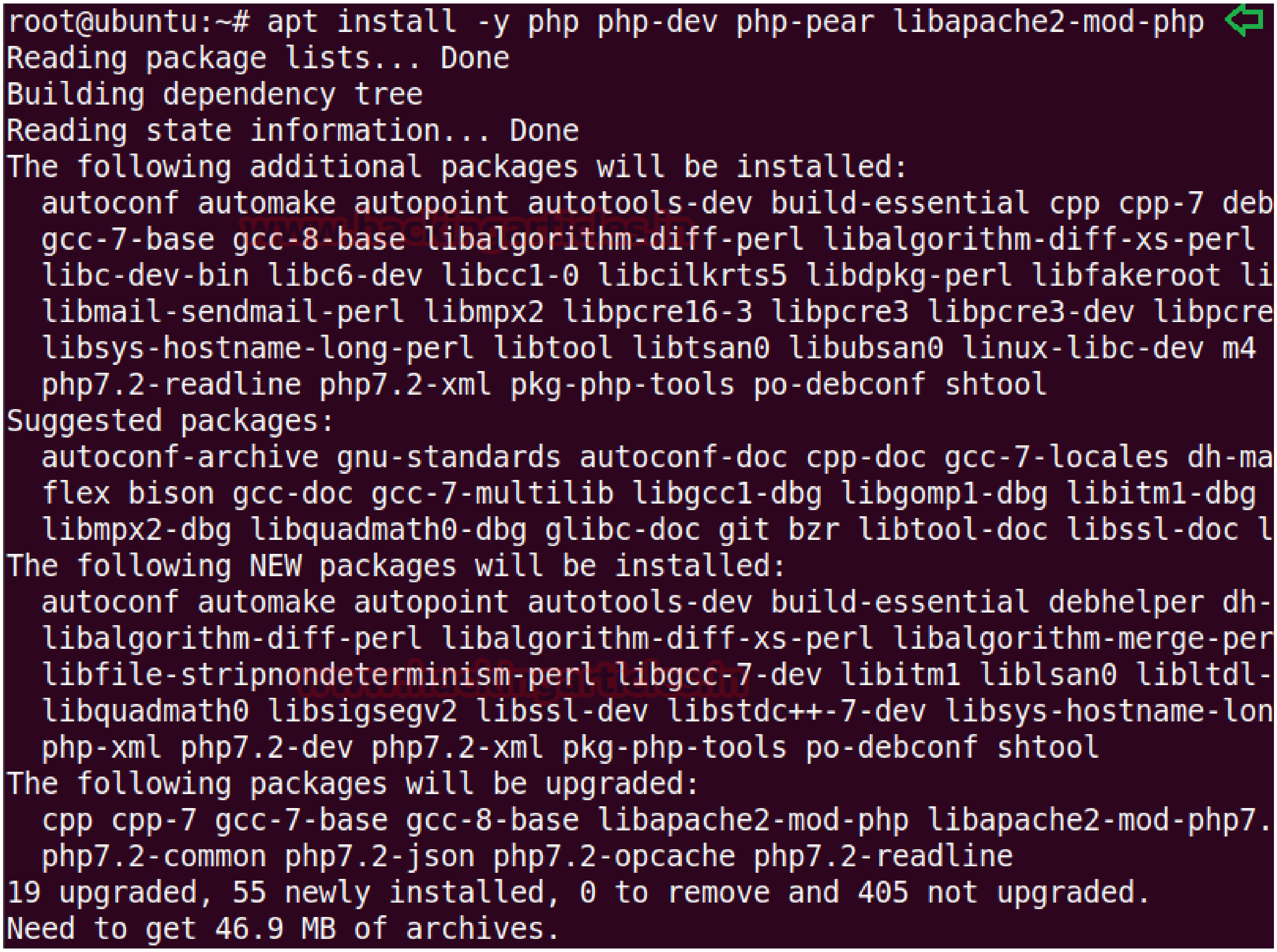
```
apt update
```



```
root@ubuntu:~# apt update
Get:1 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu bionic InRelease
Get:3 http://us.archive.ubuntu.com/ubuntu bionic-updates InRelease
Get:4 http://us.archive.ubuntu.com/ubuntu bionic-backports InRelea
Hit:5 http://ppa.launchpad.net/ondrej/php/ubuntu bionic InRelease
Fetched 252 kB in 25s (9,892 B/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
424 packages can be upgraded. Run 'apt list --upgradable' to see t
root@ubuntu:~#
```

Now, install PHP by executing the command shown below :

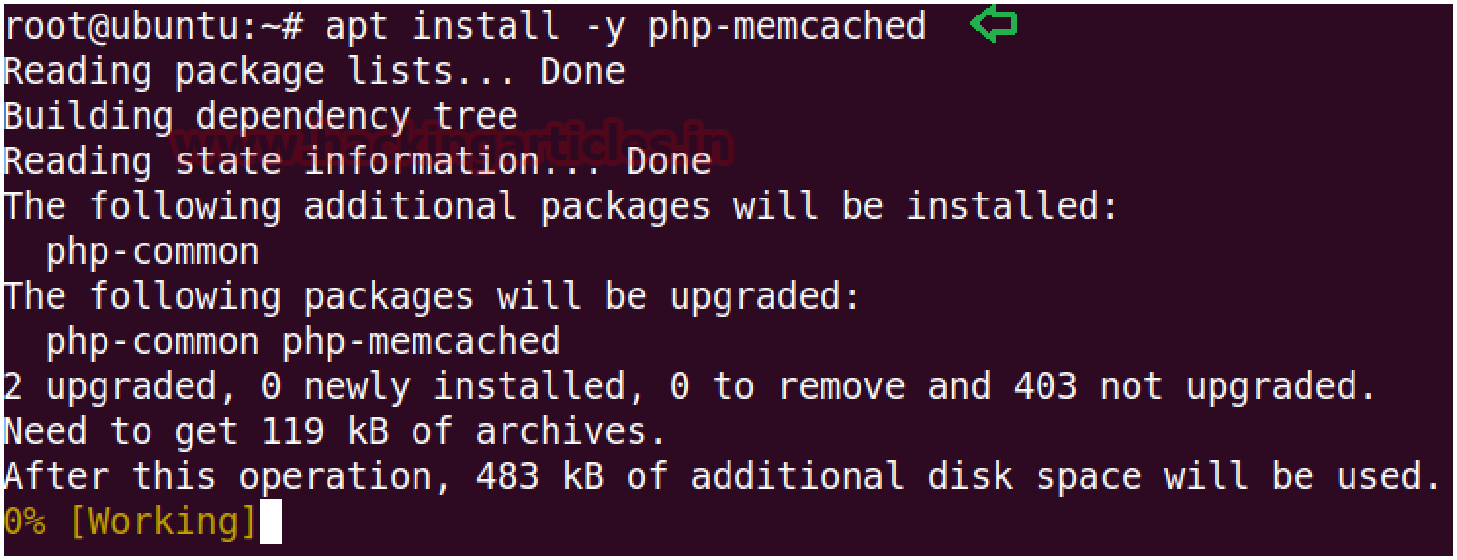
```
apt install –y php php-dev php-pear libapache2-mod-php
```



```
root@ubuntu:~# apt install -y php php-dev php-pear libapache2-mod-php
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  autoconf automake autopoint autotools-dev build-essential cpp cpp-7 deb
  gcc-7-base gcc-8-base libalgorithm-diff-perl libalgorithm-diff-xs-perl
  libc-dev-bin libc6-dev libcc1-0 libcilkrts5 libdpkg-perl libfakeroot li
  libmail-sendmail-perl libmpx2 libpcre16-3 libpcre3 libpcre3-dev libpcre
  libsys-hostname-long-perl libtool libtsan0 libubsan0 linux-libc-dev m4
  php7.2-readline php7.2-xml pkg-php-tools po-debconf shtool
Suggested packages:
  autoconf-archive gnu-standards autoconf-doc cpp-doc gcc-7-locales dh-ma
  flex bison gcc-doc gcc-7-multilib libgcc1-dbg libgomp1-dbg libitm1-dbg
  libmpx2-dbg libquadmath0-dbg glibc-doc git bzr libtool-doc libssl-doc l
The following NEW packages will be installed:
  autoconf automake autopoint autotools-dev build-essential debhelper dh-
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-per
  libfile-stripnondeterminism-perl libgcc-7-dev libitm1 liblsan0 libltdl-
  libquadmath0 libsigsegv2 libssl-dev libstdc++-7-dev libsys-hostname-lon
  php-xml php7.2-dev php7.2-xml pkg-php-tools po-debconf shtool
The following packages will be upgraded:
  cpp cpp-7 gcc-7-base gcc-8-base libapache2-mod-php libapache2-mod-php7.
  php7.2-common php7.2-json php7.2-opcache php7.2-readline
19 upgraded, 55 newly installed, 0 to remove and 405 not upgraded.
Need to get 46.9 MB of archives.
```

Now that PHP has been installed successfully in our system, we will go ahead and install the PHP Memcached module by executing the below command:

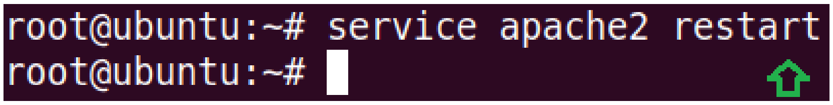
```
apt install -y php-memcached
```



```
root@ubuntu:~# apt install -y php-memcached
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  php-common
The following packages will be upgraded:
  php-common php-memcached
2 upgraded, 0 newly installed, 0 to remove and 403 not upgraded.
Need to get 119 kB of archives.
After this operation, 483 kB of additional disk space will be used.
0% [Working]
```

Once the installation is complete, restart the Apache2 service.

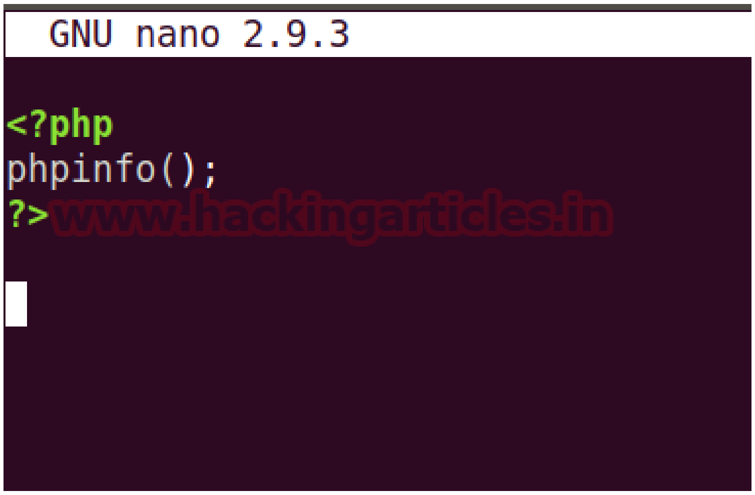
```
service apache2 restart
```



```
root@ubuntu:~# service apache2 restart
root@ubuntu:~#
```

Now check whether the PHP extension is working fine or not by creating an info.php by using the code mentioned below with nano or any text editor you like.

```
<?php
phpinfo();
?>
```



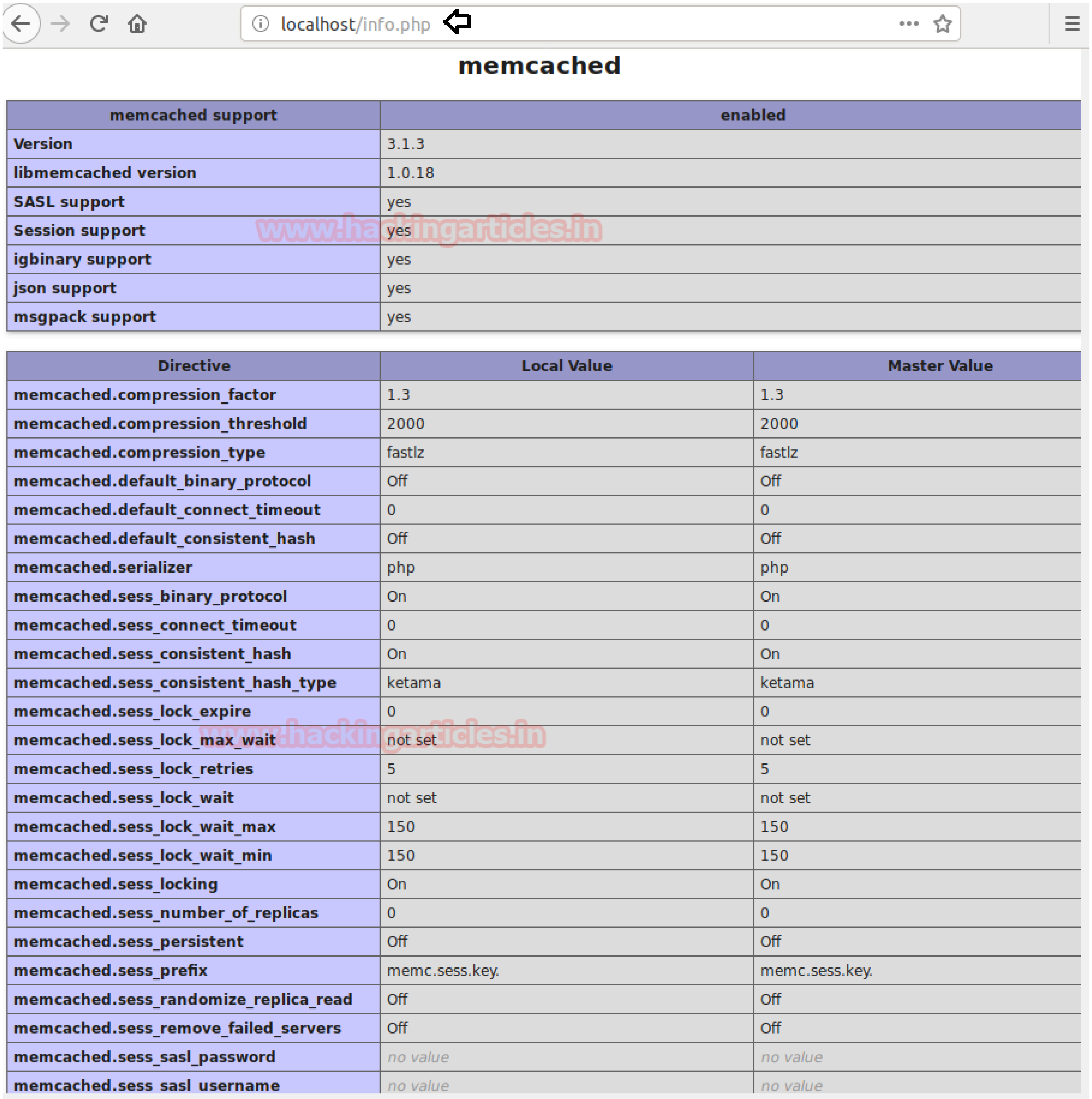
```
  GNU nano 2.9.3

<?php
phpinfo();
?>
```

Now save the file in **/var/www/html**

Once the file is saved, access it from your web browser by typing in the following URL.

```
localhost/info.php
```

You should see the results as shown in the image below.



memcached

memcached support	enabled
Version	3.1.3
libmemcached version	1.0.18
SASL support	yes
Session support	yes
igbinary support	yes
json support	yes
msgpack support	yes

Directive	Local Value	Master Value
memcached.compression_factor	1.3	1.3
memcached.compression_threshold	2000	2000
memcached.compression_type	fastlz	fastlz
memcached.default_binary_protocol	Off	Off
memcached.default_connect_timeout	0	0
memcached.default_consistent_hash	Off	Off
memcached.serializer	php	php
memcached.sess_binary_protocol	On	On
memcached.sess_connect_timeout	0	0
memcached.sess_consistent_hash	On	On
memcached.sess_consistent_hash_type	ketama	ketama
memcached.sess_lock_expire	0	0
memcached.sess_lock_max_wait	not set	not set
memcached.sess_lock_retries	5	5
memcached.sess_lock_wait	not set	not set
memcached.sess_lock_wait_max	150	150
memcached.sess_lock_wait_min	150	150
memcached.sess_locking	On	On
memcached.sess_number_of_replicas	0	0
memcached.sess_persistent	Off	Off
memcached.sess_prefix	memc.sess.key.	memc.sess.key.
memcached.sess_randomize_replica_read	Off	Off
memcached.sess_remove_failed_servers	Off	Off
memcached.sess_sasl_password	*no value*	*no value*
memcached.sess_sasl_username	*no value*	*no value*

Memcached Configuration

Now, here we are going to configure the Memcached Server. To do so, we have to edit its configuration file. You will find this configuration file through **/etc/memcached.conf** path. Open the **memcached.conf** file using nano or any other text editor. The commands that are shown below will be given and activated by default. The purpose of this mentioning is to let you know that where you can find it; along with why and how to make changes to it., if necessary. Following are the said commands which are important for low-level Memcached Server configuration:

-m 64

Here,

–m: specifies the maximum memory limit which is used by Memcached daemon. By default, this limit is 64 MB

-p 11211

Here,

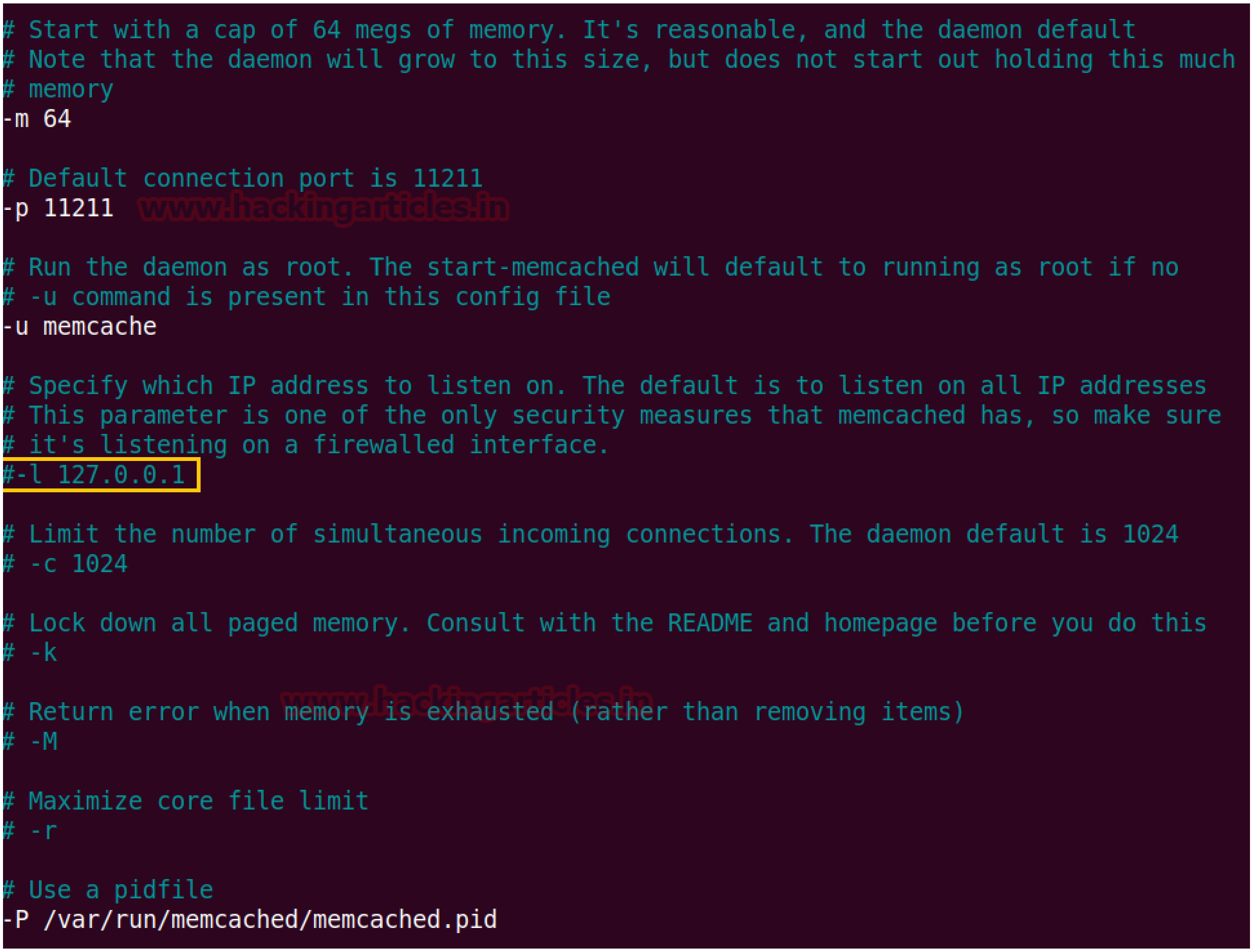
–p: specifies the port number. By default, it's 11211.

-u memcache

Here,

–u starts the daemon tool as root.

After this, uncomment "**-l 127.0.0.1**" by simply adding # as shown in the image, as it will not be so by default. By uncommenting, it will stop binding the IP address of Memcached listener to the loopback IP. Hence, traffic can come from any IP over the internet.



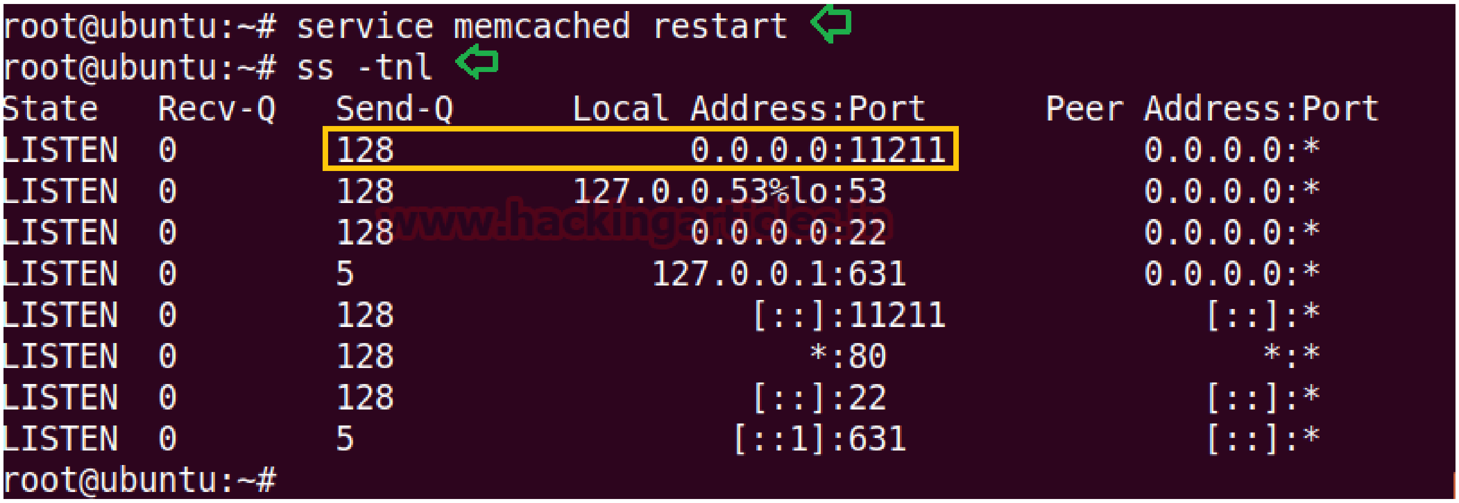
```
# Start with a cap of 64 megs of memory. It's reasonable, and the daemon default
# Note that the daemon will grow to this size, but does not start out holding this much
# memory
-m 64

# Default connection port is 11211
-p 11211

# Run the daemon as root. The start-memcached will default to running as root if no
# -u command is present in this config file
-u memcache

# Specify which IP address to listen on. The default is to listen on all IP addresses
# This parameter is one of the only security measures that memcached has, so make sure
# it's listening on a firewalled interface.
#-l 127.0.0.1

# Limit the number of simultaneous incoming connections. The daemon default is 1024
# -c 1024

# Lock down all paged memory. Consult with the README and homepage before you do this
# -k

# Return error when memory is exhausted (rather than removing items)
# -M

# Maximize core file limit
# -r

# Use a pidfile
-P /var/run/memcached/memcached.pid
```

Now once you saved the configuration file after making the changes, restart the service by using the following command :

```
service memcached restart
```

Then use the following command to confirm whether Memcached configurations are working are not :

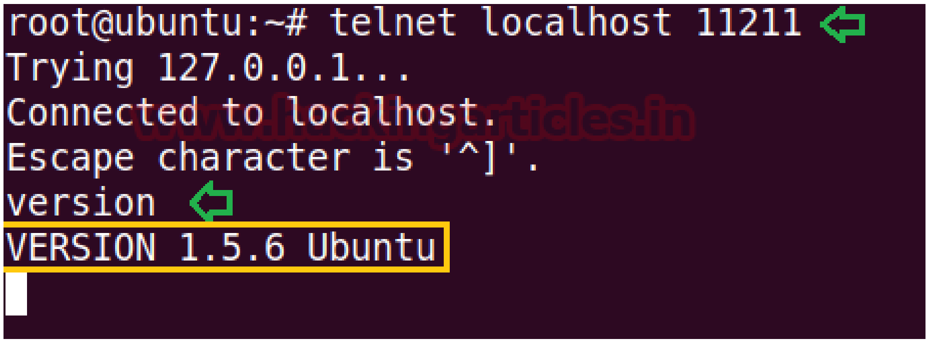
```
ss -tnl
```



```
root@ubuntu:~# service memcached restart
root@ubuntu:~# ss -tnl
State    Recv-Q    Send-Q       Local Address:Port       Peer Address:Port
LISTEN   0         128                0.0.0.0:11211           0.0.0.0:*
LISTEN   0         128          127.0.0.53%lo:53              0.0.0.0:*
LISTEN   0         128                0.0.0.0:22              0.0.0.0:*
LISTEN   0         5                127.0.0.1:631             0.0.0.0:*
LISTEN   0         128                   [::]:11211              [::]:*
LISTEN   0         128                      *:80                    *:*
LISTEN   0         128                   [::]:22                 [::]:*
LISTEN   0         5                    [::1]:631                [::]:*
root@ubuntu:~#
```

Once you are done with the above commands, connect Memcached through telnet and do a version check by typing in “version” command:

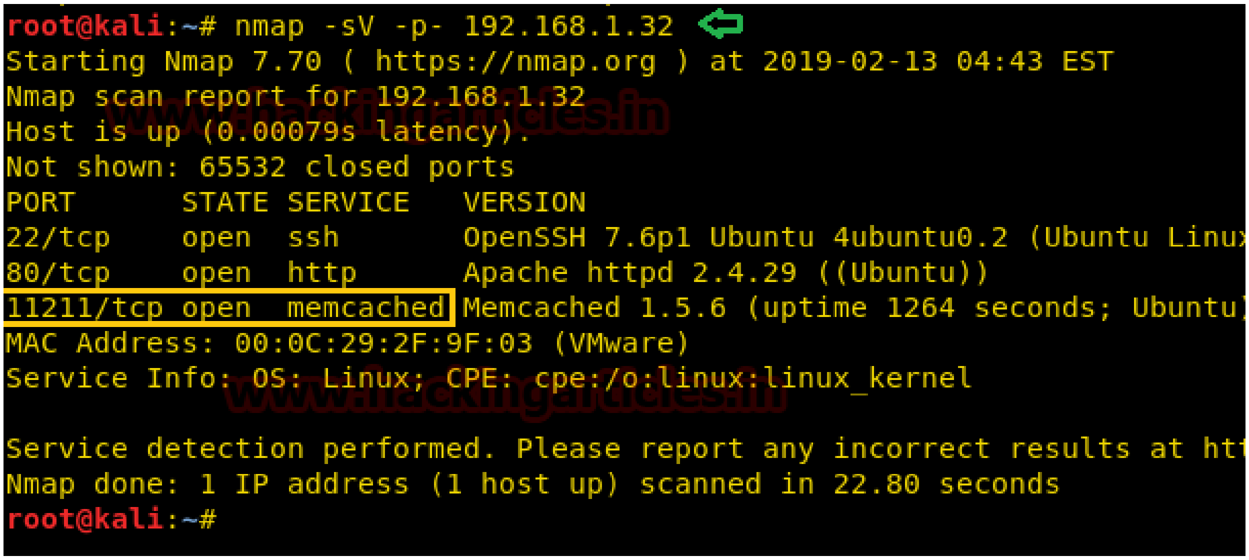
```
telnet localhost 11211
version
```



```
root@ubuntu:~# telnet localhost 11211
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
version
VERSION 1.5.6 Ubuntu
```

Now, you can use nmap to check if the Memcached service is running on the server.

```
nmap -sV -p- 192.168.1.32
```



```
root@kali:~# nmap -sV -p- 192.168.1.32
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-13 04:43 EST
Nmap scan report for 192.168.1.32
Host is up (0.00079s latency).
Not shown: 65532 closed ports
PORT      STATE SERVICE   VERSION
22/tcp    open  ssh       OpenSSH 7.6p1 Ubuntu 4ubuntu0.2 (Ubuntu Linux
80/tcp    open  http      Apache httpd 2.4.29 ((Ubuntu))
11211/tcp open  memcached Memcached 1.5.6 (uptime 1264 seconds; Ubuntu)
MAC Address: 00:0C:29:2F:9F:03 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at htt
Nmap done: 1 IP address (1 host up) scanned in 22.80 seconds
root@kali:~#
```

Conclusion

To conclude, we can say that Memcached is a distributed memory caching system. It uses expiration timeouts i.e. if the server has no memory left, it will evict items to replace them with the new ones. The items it chooses to replace are the ones which have not been requested for a long period of time. And so, in the above article, we have provided a basic guide to set up the Memcached penetration testing lab.