Installing Memcached on Windows

Submitted by COMMASTER

15

Memcached is a high performance, in-memory key-value store or caching system. Its main purpose is to speed up web applications by caching database queries, contents, or other computed results.

Memcached is originally a linux application, but since it is open-source, it has been compiled for windows. There are two major sources for the pre-built windows binary: Jellycan and Northscale, and both versions can be used. The following are the download links for the memcached windows binaries:

http://code.jellycan.com/files/memcached-1.2.5-win32-bin.zip
(http://code.jellycan.com/files/memcached-1.2.5-win32-bin.zip)
http://code.jellycan.com/files/memcached-1.2.6-win32-bin.zip
(http://code.jellycan.com/files/memcached-1.2.6-win32-bin.zip)
http://downloads.northscale.com/memcached-win32-1.4.4-14.zip
(http://downloads.northscale.com/memcached-win32-1.4.4-14.zip)
http://downloads.northscale.com/memcached-win64-1.4.4-14.zip
(http://downloads.northscale.com/memcached-win64-1.4.4-14.zip)
http://downloads.northscale.com/memcached-1.4.5-x86.zip
(http://downloads.northscale.com/memcached-1.4.5-x86.zip)
http://downloads.northscale.com/memcached-1.4.5-amd64.zip
(http://downloads.northscale.com/memcached-1.4.5-amd64.zip)

In versions earlier than 1.4.5, memcached can install itself as a service. However, the ability to run memcached as a service is removed since version 1.4.5. Therefore, the installation steps are divided into two categories, part A for memcached prior to version 1.4.5. and part B for memcached version 1.4.5 and later.

A) Installation of memcached < 1.4.5:

1. Extract the memcached windows binary to any directory.
2. In versions earlier than 1.4.5, memcached can install itself as a service. Run a command prompt with **elevated privileges (http://kb.macrium.com/KnowledgebaseArticle50132.aspx)**, and type:

```
c:\memcached\memcached.exe -d install
```

   * Replace *c:\memcached\memcached.exe* with the actual path of your installation.

3. Then, start or stop the memcached service with the following command:

```
c:\memcached\memcached.exe -d start
c:\memcached\memcached.exe -d stop
```

4. To change the configuration of memcached, run *regedit.exe* and navigate to the key "*HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\memcached*". Suppose you wish to increase the memory limit of memcached, edit the value of *ImagePath* to the following:

```
"c:\memcached\memcached.exe" -d runservice -m 512
```

   * Besides '*-m 512*', you may also append other memcached parameters to the path. Run "*c:\memcached\memcached.exe -h*" to view the list of available parameters.

5. Meanwhile, to uninstall the memcached serivce, run the following command:

```
c:\memcached\memcached.exe -d uninstall
```

B) Installation of memcached >= 1.4.5:

1. Extract the memcached windows binary to any directory.
2. In version 1.4.5 or later, memcached cannot run as a service. It must be started as a normal process using the task scheduler. To configure the memcached process to run automatically every time windows start, run a command prompt with **elevated privileges (http://kb.macrium.com/KnowledgebaseArticle50132.aspx)** , and type the following:

```
schtasks /create /sc onstart /tn memcached /tr "'c:\memcached\memcached.exe' -m 512"
```

   * Replace *c:\memcached\memcached.exe* with the actual path of your installation.
   ** Besides '*-m 512*', you may also append other memcached parameters to the path. Run "*c:\memcached\memcached.exe -h*" to view the list of available parameters.

3. Meanwhile, to remove the scheduled memcached task, run the following command:

```
schtasks /delete /tn memcached
```

Integrating with PHP

To interface with memcached in PHP, you need to install the memcache extension for PHP:

1. Check that your PHP extension folder has the file php_memcache.dll. If not, download the file from **https://pecl.php.net/package/memcache** (select the windows dll file), and place it in the PHP extension folder.
2. Add the following line in php.ini to enable the memcache extension.
   extension=php_memcache.dll
3. Create this simple php script file to test that it works.

```
<?php

$memcache = new Memcache;
$memcache->connect('localhost', 11211) or die ("Could not connect");

$version = $memcache->getVersion();
echo "Server's version: ".$version."<br/>\n";

$tmp_object = new stdClass;
$tmp_object->str_attr = 'test';
$tmp_object->int_attr = 123;

$memcache->set('key', $tmp_object, false, 10) or die ("Failed to save data at the server
echo "Store data in the cache (data will expire in 10 seconds)<br/>\n";

$get_result = $memcache->get('key');
echo "Data from the cache:<br/>\n";

var_dump($get_result);

?>
```

Integrating with Python

To interface with memcached in Python, you need to install the memcached client for Python.

1. Execute one of the following command to install the memcached client. The first is for Python 2.x while the second is for Python 3.x.

```
pip install python-memcached
pip install python3-memcached
```

2. Create this simple python script to test that it works.

```
import memcache
mc = memcache.Client(['127.0.0.1:11211'], debug=0)
mc.set("some_key", "Some value")
value = mc.get("some_key")
mc.set("another_key", 3)
mc.delete("another_key")
mc.set("key", "1")   # note that the key used for incr/decr must be a string.
mc.incr("key")
mc.decr("key")
```

Memcached statistics

To view the statistics for memcached, bring up a telnet connection to memcached by the command:

```
telnet 127.0.0.1 11211
```

Then, type *stats* and enter.

Here is an explanation of the different memcached stats.

Name	Type	Meaning
pid	32u	Process id of this server process
uptime	32u	Number of secs since the server started
time	32u	current UNIX time according to the server
version	string	Version string of this server
pointer_size	32	Default size of pointers on the host OS (generally 32 or 64)
rusage_user	32u.32u	Accumulated user time for this process (seconds:microseconds)
rusage_system	32u.32u	Accumulated system time for this process (seconds:microseconds)
curr_items	32u	Current number of items stored
total_items	32u	Total number of items stored since the server started
bytes	64u	Current number of bytes used to store items
curr_connections	32u	Number of open connections
total_connections	32u	Total number of connections opened since the server started running
connection_structures	32u	Number of connection structures allocated by the server
reserved_fds	32u	Number of misc fds used internally
cmd_get	64u	Cumulative number of retrieval reqs
cmd_set	64u	Cumulative number of storage reqs

Name	Type	Meaning
cmd_flush	64u	Cumulative number of flush reqs
cmd_touch	64u	Cumulative number of touch reqs
get_hits	64u	Number of keys that have been requested and found present
get_misses	64u	Number of items that have been requested and not found
delete_misses	64u	Number of deletions reqs for missing keys
delete_hits	64u	Number of deletion reqs resulting in an item being removed.
incr_misses	64u	Number of incr reqs against missing keys.
incr_hits	64u	Number of successful incr reqs.
decr_misses	64u	Number of decr reqs against missing keys.
decr_hits	64u	Number of successful decr reqs.
cas_misses	64u	Number of CAS reqs against missing keys.
cas_hits	64u	Number of successful CAS reqs.
cas_badval	64u	Number of CAS reqs for which a key was found, but the CAS value did not match.
touch_hits	64u	Numer of keys that have been touched with a new expiration time
touch_misses	64u	Numer of items that have been touched and not found
auth_cmds	64u	Number of authentication commands handled, success or failure.
auth_errors	64u	Number of failed authentications.
evictions	64u	Number of valid items removed from cache to free memory for new items
reclaimed	64u	Number of times an entry was stored using memory from an expired entry
bytes_read	64u	Total number of bytes read by this server from network
bytes_written	64u	Total number of bytes sent by this server to network
limit_maxbytes	32u	Number of bytes this server is allowed to use for storage.
threads	32u	Number of worker threads requested. (see doc/threads.txt)
conn_yields	64u	Number of times any connection yielded to another due to hitting the -R limit.
hash_power_level	32u	Current size multiplier for hash table
hash_bytes	64u	Bytes currently used by hash tables
hash_is_expanding	bool	Indicates if the hash table is being grown to a new size
expired_unfetched	64u	Items pulled from LRU that were never touched by get/incr/append/etc before expiring
evicted_unfetched	64u	Items evicted from LRU that were never touched by get/incr/append/etc.
slab_reassign_running	bool	If a slab page is being moved
slabs_moved	64u	Total slab pages moved
crawler_reclaimed	64u	Total items freed by LRU Crawler
lrutail_reflocked	64u	Times LRU tail was found with active ref. Items moved to head to avoid OOM errors.
Name	***Type***	***Meaning***