OmniTier

MemStac™: FREQUENTLY ASKED QUESTIONS

Q. What is Memcached?

Memcached is a widely used distributed, in-memory caching solution for database caching.

Q. What is MemStac™?

MemStac™ is a Memcached compatible key-value solution that provides a much higher cache capacity at significantly lower cost than Memcached. It achieves higher capacity by using tiered-memory based on DRAM and NVMe SSD.

Q. What are key benefits of MemStac?

Modern databases are several orders of magnitude slower than either MemStac™ or Memcached – tens of thousands of requests per second compared to millions. Since overall system performance is determined not only by the speed of the cache but the percentage of requests serviced by the cache (hit rate), insufficient cache capacity can cause the database to be the primary performance bottleneck. Due to its inherent capacity limitation, Memcached may only be able to service a small fraction of the system workload. This causes most of the requests to be directed to the database, leading to poor system performance.

MemStac™, on the other hand, with its significantly larger cache size, can economically cache a majority of the application's requests and reduce the load on the backend database, resulting in a system-level performance improvement of up to 80x. MemStac™ on AWS i3 instances can provide up to 30x higher capacity at 30x lower $/GB/Hour compared to Memcached running EC2 r4 instances.

Q. How to launch MemStac™ on AWS?

a. Go to the MemStac™ AWS marketplace page
b. Press "Continue to subscribe" button
c. Select region
d. Select EC2 Instance type

e. Select security groups
    a. Configure ports 11211, 6060 and 22 to be accessible to your Memcached clients

f. Add SSH Keypair for SSH access
g. Press "Launch with 1-Click" button

Q. How to SSH to EC2 instance?

a. Go to your EC2 dashboard.
b. Get the IP address of the newly created MemStac™ instance
c. Use a SSH client to login into the EC2 instance (username: ec2-user). If you can't SSH to the EC2 instance, please check the security group settings and SSH keypair

Q. What are the ports used by MemStac™ server?

MemStac™ uses TCP port 11211 to process Memcached commands and TCP port 6060 for monitoring.

Q. How to check that the MemStac™ service is running?

a. SSH to the EC2 instance where MemStac™ is running
b. Run the following command to check if the MemStac™ process is up and running
c. If you don't see the MemStac™ process running, please contact support@omnitier.com

```
[ec2-user@ip-172-31-51-151 ~]$ ps aux | grep memstac  | grep -v grep

root     3820 74.0 69.5 235619984 175116032 ? Sl   19:59   2:52 /opt/omnitier/bin/memstac
```


Q. How to monitor MemStac™?

To monitor MemStac™, you can use a CURL command to get metrics in JSON format. For more information on MemStac™ monitoring, please refer to the MemStac™ user guide.

```
curl http://<MemStac EC2 IP address>:6060/metrics
```

Example:

```
[ec2-user@ip-172-31-11-111 ~]$ curl http://172.31.11.112:6060/metrics
{
  "TotalNumConnections": 0,
  "CurrentNumConnections": 0,
  "PerOpcodeMetrics": {},
  "NumItems": 0,
  "NumEvictedItems": 0,
  "DiskErrors": null,
```

```
    "SmartLogs": null,
    "CpuUsage": {
        "UserPct": 0,
        "NicePct": 0,
        "SystemPct": 0,
        "IdlePct": 0,
        "IowaitPct": 0,
        "IrqPct": 0,
        "SoftIrqPct": 0,
        "StealPct": 0,
        "GuestPct": 0,
        "GuestNicePct": 0,
        "Seconds": 0
    },
    "MemUsage": {
        "Mem_total": 0,
        "Mem_used": 0,
        "Mem_free": 0,
        "Buffers": 0,
        "Cached": 0,
        "Swap_total": 0,
        "Swap_used": 0,
        "Swap_free": 0
    }
}
```

Q. How to troubleshoot if you are not able to connect your application to MemStac™?

You can use telnet to access MemStac™. To simplify troubleshooting, we recommend running this test on the host that has MemStac™ running. First, SSH to the EC2 instance where MemStac™ is running. If you are having trouble in SSH to the instance, please refer to the section "**How to SSH to EC2 instance?".** Second, if you are using Amazon Linux, use "**sudo yum install telnet**" command to install telnet, once you've successfully ssh to the host. You can remove telnet by "**sudo yum remove telnet**" when you have finished testing.

Screenshot below is an example of the MemStac™ telnet test.

```
[ec2-user@ip-172-31-80-192 ~]$ telnet localhost 11211
Trying 127.0.0.1...
Connected to localhost.
```

```
Escape character is '^]'.
set foo 0 0 3
bar
STORED
get foo
VALUE foo 0 3
bar
END
stats
STAT pid 2734
...
```

- "**telnet localhost 11211**" is the telnet command to access the MemStac™ service. If you see the "Connected to localhost" message, then the MemStac™ service is running and you are ready to execute more commands.
- By typing "**set foo 0 0 3**" and then "**bar**", you are storing a key:foo/value:bar pair in MemStac™. It will print "STORED" prompt when it is done.
- By typing "**get foo**", you are extracting a value from MemStac with a key:foo. It will print "VALUE foo 0 3" and following "bar" as the value.
- You can check general-purpose statistics by typing "**stats**" command. Statistics like pid, uptime, version, etc. are available.
- Hit **[Ctrl]+]** to disconnect from the port 11211, and type "**quit**" to end telnet session.

If the above commands work, the next step is to make sure your security group configuration allows MemStac™ server to receive request on TCP port 11211 from your application host. To confirm, repeat the telnet test method above from your application host to MemStac™ server's IP address instead of localhost, (e.g. telnet 172.31.80.192 11211).

Please check if you are using the ASCII version of Memcached client library and follow the instructions provided by the client library.

Q. How to test MemStac™ with a Memcached client?

a. Make sure PHP and Memcached client libraries are installed on the PHP application server
b. Make sure MemStac™ server's security group is configured to allow inbound traffic from PHP application server on TCP port 11211
c. Here is a simple PHP script that connects to MemStac™ server using the PHP Memcached client library.

```php
<?php

$mc = new Memcached();
$mc->addServer("<MemStac  EC2 IP address>", 11211);

$mc->set("foo", "Hello!");
```

```
$mc->set("bar", "World! ");

$arr = array(
    $mc->get("foo"),
    $mc->get("bar")
);
var_dump($arr);
?>
```

d. The above script stores two keys "foo" and "bar" into MemStac™ and retrieves both keys and displays their values. To verify this behavior, please run:

```
$  php  simple.php
```

Q. What Memcached client libraries were tested with MemStac™?

MemStac™ is fully compatible with Memcached ASCII protocol and requires no application level or client library level changes to interface with MemStac™. The following section describes the list of client libraries that have been tested with MemStac™:

Metric	Description
C/C++	http://libmemcached.org
C#	https://github.com/enyim/EnyimMemcached
Go	https://github.com/bradfitz/gomemcache
Java	https://github.com/couchbase/spymemcached https://github.com/gwhalin/Memcached-Java-Client https://github.com/spotify/folsom
Node.js	https://github.com/3rd-Eden/memcached
PHP	https://github.com/php-memcached-dev/php-memcached
Python	https://github.com/pinterest/pymemcache
Ruby	https://github.com/petergoldstein/dalli

Q. How to contact OmniTier?

Email your support requests to support@omnitier.com

About OmniTier

OmniTier Inc., founded in 2015, develops and supports integrated software solutions for memory-centric infrastructure applications, including high performance object caching, scientific analysis for machine learning, AI, and genomics. Its leadership team has a track record of delivering many industry firsts in data storage and access across different media types. The company has offices in Milpitas, California, and Rochester, Minnesota