

# Bug Report

cd-037

Aryamman Jain, Pranav Mehndiratta, Vaibhav Vijay

## Bug 1

Bug Title	Usage of strtok() for parsing
Date Seen	March 12, 2014
Environment	Ubuntu 13.10 x64
Description	Strings parsed by strtok are modified, cannot be reused in other functions.
Severity*	Major
Steps to Reproduce	1) Connect to server and authenticate. 2) Set a new key and value using the client shell. 3) Get the same key through client shell
Actual Behavior	Printed value is only the first column-value pair though record contains more than 1 column.
Expected Behavior	Complete value i.e. all column-value pairs are printed when accessed through a Get request.
Troubleshooting	1) Check length of returned value and print that many characters.
Solution	Run strtok() on a copy of the input using strcpy().

## Bug 2

Bug Title	Setting of authenticated "boolean flag" back to false in "client_disconnect"
Date Seen	March 14, 2014
Environment	Mac OSX
Description	Segment faults, as it required access to variables that required authentication.
Severity*	Trivial
Steps to Reproduce	1) Connect to server and authenticate. 2) Setting a key with value 3) Disconnect from server 4) Getting stored key
Actual Behavior	Incorrect error code. Printed segment fault when disconnected.
Expected Behavior	After disconnecting, server shouldn't allow get/set and print the correct error code.
Troubleshooting	1) Printing error code and adding printing flags in client_disconnect().
Solution	Change the 'authenticated' boolean in client_disconnect() back to false.

\*Severity ranking: Trivial, Minor, Major or Catastrophic

### Bug 3 (Figure 1)

Bug Title	Incorrect result for number of matched keys from query func
Date Seen	March 14, 2014
Environment	SMP Debian 3.2.54 x86_64
Description	Number of keys were being incremented on modification and addition of records
Severity*	Major
Steps to Reproduce	<ol style="list-style-type: none"> <li>1) Connect to server and authenticate.</li> <li>2) Setting a key with value</li> <li>3) Setting same key with new value (repeat many times)</li> <li>4) Running query that should return this key</li> </ol>
Actual Behavior	Same key is returned multiple times
Expected Behavior	Only one key should be returned (as keys are unique)
Troubleshooting	1) Print all keys being compared to predicate
Solution	Problem traced back to server_set. Number of keys were being incremented during both addition/modification of records.

```

^
ar rcs libstorage.a storage.o utils.o
cc -g -Wall -c server.c -o server.o
cc -g -Wall server.o utils.o -o server -lcrypt
cc -g -Wall -c client.c -o client.o
cc -g -Wall client.o libstorage.a -o client -lcrypt
cc -g -Wall -c encrypt_passwd.c -o encrypt_passwd.o
cc -g -Wall encrypt_passwd.o utils.o -o encrypt_passwd
pranav@Pranav-Notebook:~/Desktop/storage/src$ ./server d
server main: Server on localhost:1111
server main: Got a connection from 127.0.0.1:653
handle_command: Processing command 'AUTH #admin #xxGH.5Z
server main: Closed connection from 127.0.0.1:653
server main: Got a connection from 127.0.0.1:1677
handle_command: Processing command 'AUTH #admin #xxiz1FI
handle_command: Processing command 'SET #cars #BMW #bran
handle_command: Processing command 'SET #cars #Random #b
handle_command: Processing command 'QUERY #cars #2 #bran
handle_command: Processing command 'QUERY #cars #2 #prio
-----
1) Connect
2) Authenticate
3) Get
4) Set
5) Query
6) Disconnect
7) Exit
-----
Please enter your selection: 5
Please enter the table name: cars
Please enter the predicates: brand = hey
Please enter the maximum number of keys: 5
storage_query: 2 keys matched with 'brand = hey' in tabl
Matched Keys: 'BMW', 'BMW'

```

Figure 1: Captured while debugging

\*Severity ranking: Trivial, Minor, Major or Catastrophic