

```

1  //All methods ought to be static
2  //Most real parsing occurs in the Message class, this just passes commands to DB
3  package ballmerpeak.turtlenet.server;
4
5  import ballmerpeak.turtlenet.shared.Message;
6
7  public class Parser {
8      /* Useful to ID the type of message on behalf of the DB so it can use type
9       * specific get methods (e.g.: Message.PCHATgetConversationID()). Most
10      * parsing actually occurs in the Message class itself. Maybe this should
11      * be changed so parsing occurs here, e.g.: Parser.LIKEgetItemID(msg), but
12      * msg.LIKEgetItemID() is more natural.
13      */
14      public static void parse (Message msg, Database db) {
15
16          Logger.write("VERBOSE", "PARSE", "parsing message");
17
18          escape(msg);
19
20          if (db.isRevoked(db.getSignatory(msg)))
21              Logger.write("WARNING", "PARSE", "Revoked key in use, message dropped");
22          else if (msg.getCmd().equals("POST")) //post to own wall
23              db.addPost(msg);
24          else if (msg.getCmd().equals("CLAIM")) //claim a username
25              db.addClaim(msg);
26          else if (msg.getCmd().equals("REVOKE")) //revoke private key
27              db.addRevocation(msg);
28          else if (msg.getCmd().equals("PDATA")) //create or update profile data
29              db.addPDATA(msg);
30          else if (msg.getCmd().equals("CHAT")) //establish chat
31              db.addConvo(msg);
32          else if (msg.getCmd().equals("PCHAT")) //add message to chat
33              db.addMessageToChat(msg);
34          else if (msg.getCmd().equals("CMNT")) //comment
35              db.addComment(msg);
36          else if (msg.getCmd().equals("LIKE")) //like
37              db.addLike(msg);
38          else if (msg.getCmd().equals("UNLIKE")) //like
39              db.unlike(msg.UNLIKEgetItemID());
40          else if (msg.getCmd().equals("EVNT")) //event
41              db.addEvent(msg);
42          else if (msg.getCmd().equals("ADDCAT")) //add category
43              db.addCategory(msg);
44          else if (msg.getCmd().equals("UPDATECAT")) //update categorys canSeePDATA
45              db.updatePDATAPermission(msg);
46          else if (msg.getCmd().equals("ADDTOCAT")) //add key to category
47              db.addToCategory(msg);
48          else if (msg.getCmd().equals("REMFROMCAT")) //remove key from category
49              db.removeFromCategory(msg);
50          else if (msg.getCmd().equals("ADDKEY")) //add public key
51              db.addKey(msg);
52          else if (msg.getCmd().equals("NULL"))
53              Logger.write("VERBOSE", "PARSE", "undecryptable message"); //not for us
54          else if (!msg.getCmd().equals("FPOST"))
55              Logger.write("ERROR", "PARSE", "Unknown message type: \"" + msg.getCmd() + "\"");
56
57          if (msg.getCmd().equals("FPOST"))
58              Logger.write("WARNING", "PARSE", "FPOST is depreciated");
59      }
60
61      private static void escape (Message m) {
62          m.content = m.content.replace("'", "''");
63      }
64  }

```