UK = Public Key
RK = Public Key

Connecting to Server by Client

1.
    1.1. Client connects to Server, sending UK.
    1.2. Server sends it's UK, acting as the Certificate Authority
2.
    2.1. Server sends a certificate back encrypted with clients UK(wait why am I doing this?), and stores on Server. Certificate contains Client's UK
    2.2. On a new connection go from step 1.2. On step 2.1 do that but also distribute the new Certificate to old Clients + share all old Certificates with the new Client.

Sending Message

3. Original Message compressed and then hashed to make a Digest (D).
4. D encrypted with RK_{sender} to make Signed Digest (SD).
5. SD added to original Message (SD+MSG=/SDMSG).
6. Create One-Time Secret Key (SK_1).
7. SDMG encrypted with SK_1 (ESDMSG).
8. SK_1 encrypted with Server(?) UK (ESK)
9. ESK + ESDMSG sent to Server

[sent to Server: ESK + ESDMSG]

10. ESK decrypted with RK_{Server} to get back SK_1.
11. ESDMSG decrypted with SK_1 to get back SDMSG (SD + MSG). [Wait nah, I think this is cap]
12. For each recipient:
    12.1. SK_1 encrypted with $UK_{RecipientN}$ to get ESK_2
    12.2. ESK_2 + ESDMSG sent to Recipient N

[To Recipient N: ESK_2 + ESDMSG]

Receiving Message

13. ESK_2 decrypted with their own RK => SK_1.
14. ESDMSG decrypted with SK_1 => SDMSG.
15. SD decrypted with UK_{Sender} => D_A.
16. MSG compressed + hashed => D_B.
17. D_A and D_B are compared for integrity.