

Drip Capital - Engineering Test

Token Problem

Problem:

Write a server which can generate and assign random tokens within a pool and release them after some time. Following endpoints should work on your server:

1. Endpoint to generate unique token in the pool.
2. Endpoint to assign unique token. On hitting this endpoint, server should assign available random token from the pool and should not serve the same token again until it's freed or unblocked. If no free token is available in pool, it should serve 404.
3. Endpoint to unblock the token in the pool. Unblocked token can then be served in (2)
4. Endpoint to delete the token in the pool. Deleted token should be removed from the pool.
5. Endpoint to keep the tokens alive. All tokens should receive this endpoint within 5 minutes. If a token doesn't receive a keep-alive in last 5 mins, it should be deleted from pool and should not be served again.
6. By default each token will be freed/released automatically after 60s of use. To keep the token allocated to himself, client should request keep-alive (5) on same token within 60s.

Enforcement: No operation should result in iteration of whole set of tokens; i.e, complexity cannot be $O(n)$.