Sidejacking Attack Discussion

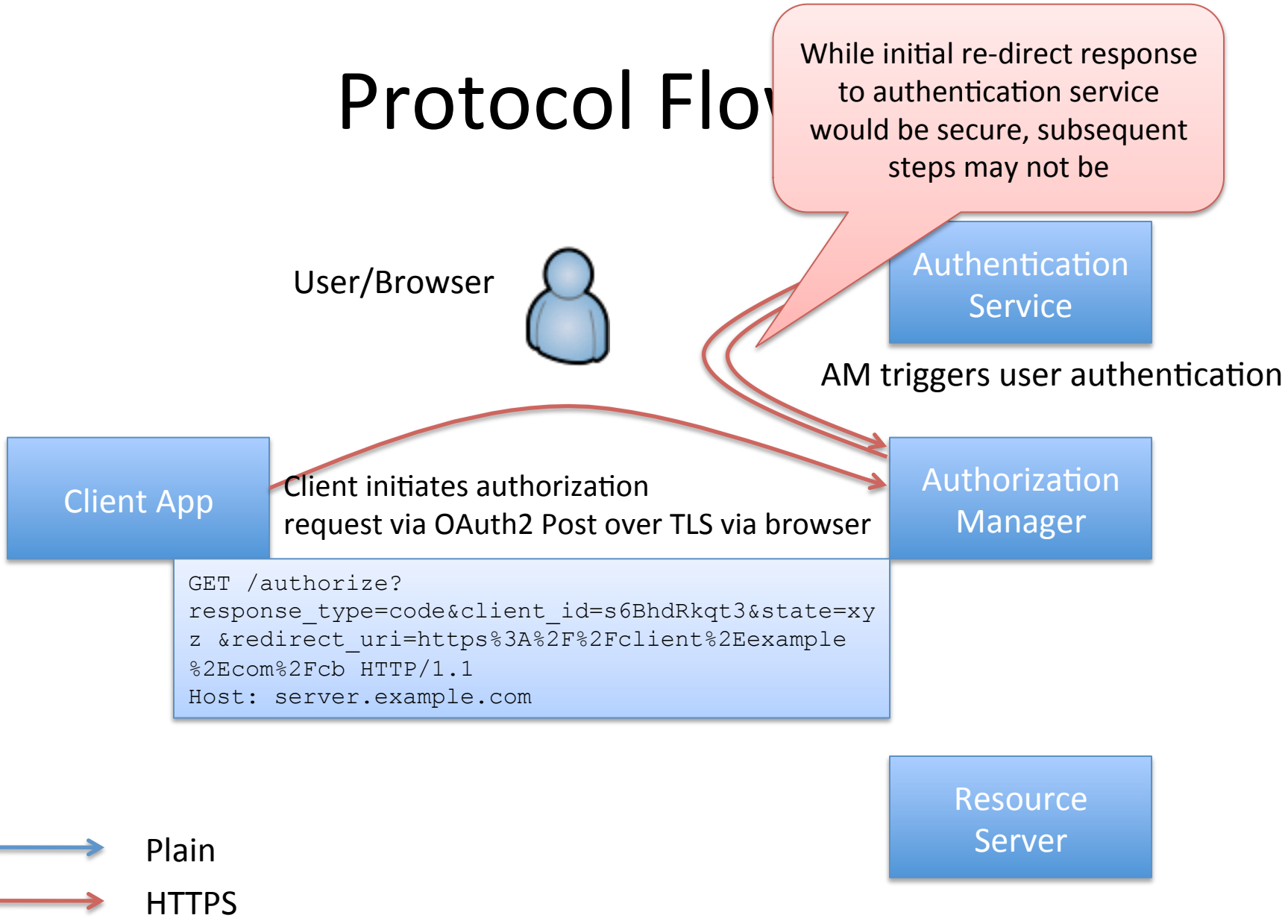
IETF 81 Meeting
Web Authorization Protocol WG

Phil Hunt
July 24, 2011

Issue

- There exists a strong risk of being able to sniff authorization codes and steal sessions by sidejacking
- There is an assumption that if an authorization server mandates TLS that the full authorization sequence is secure
  - If HTTP Request is secure then Response must also be secure – Not in practice….

Protocol Flo

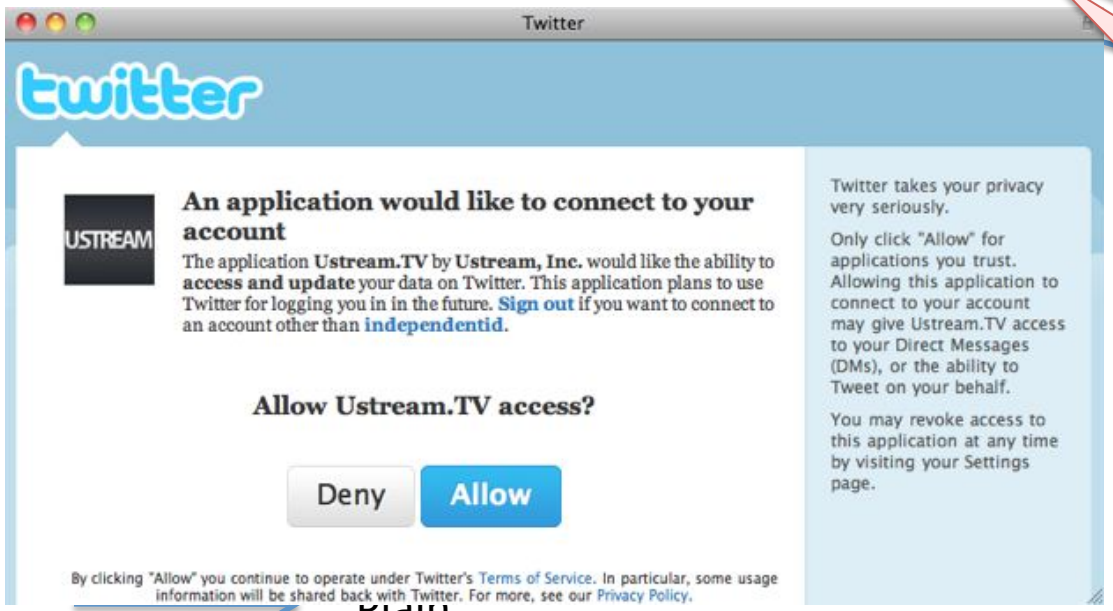


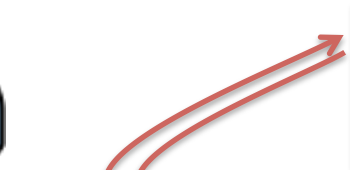
Analysis

- The authorization sequence is only secure IF
  - TLS is enabled
  - The request and response are “atomic”
    - The response follows immediately from a request
- Issue:
  - Authorization is often multi-step requiring multiple redirects
  - Flows are not atomic and involve many request response pairs.

Protocol Flow

AM asks user for consent



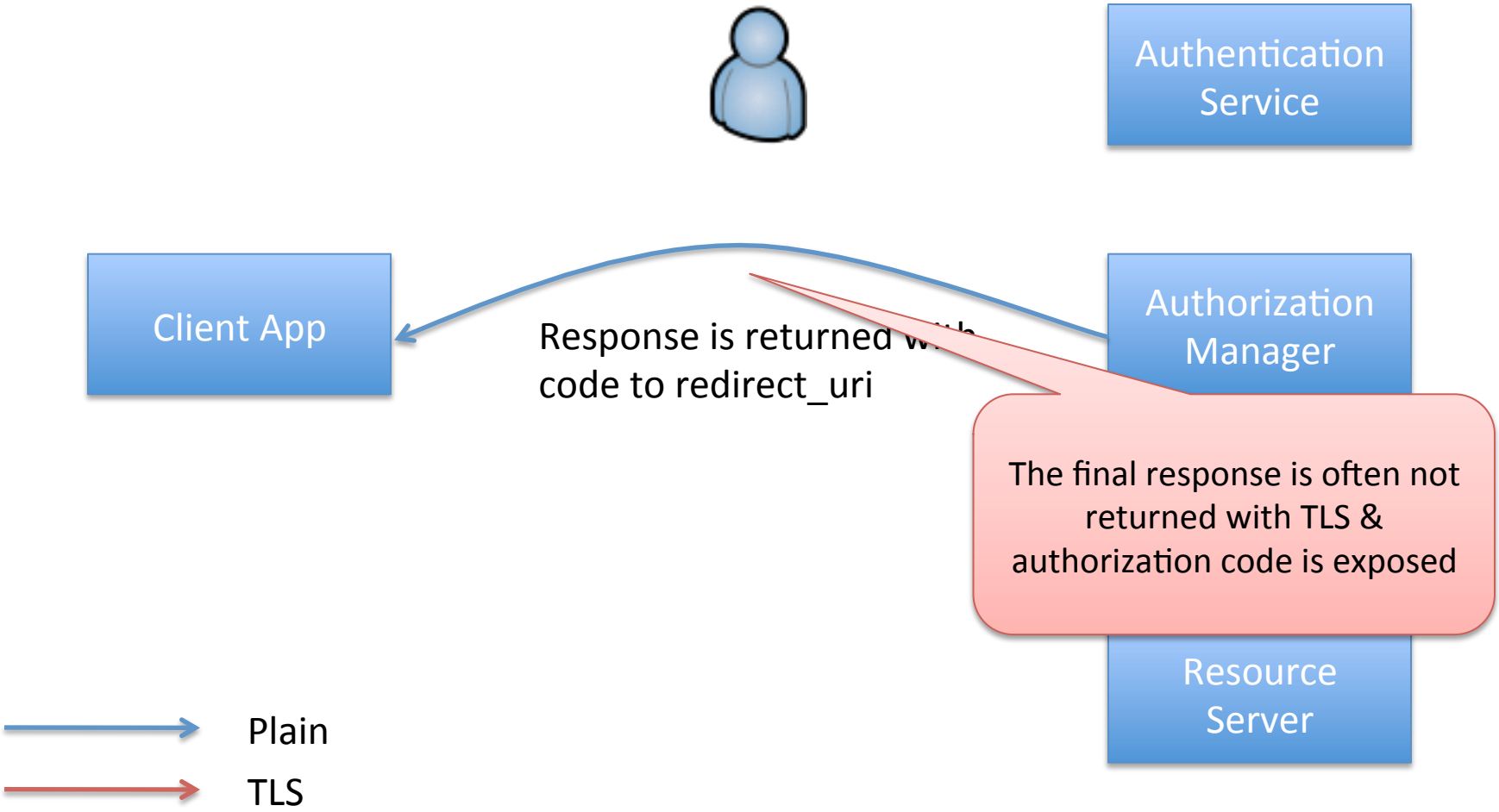


While the user is "authenticated" the authorization step is often done without TLS

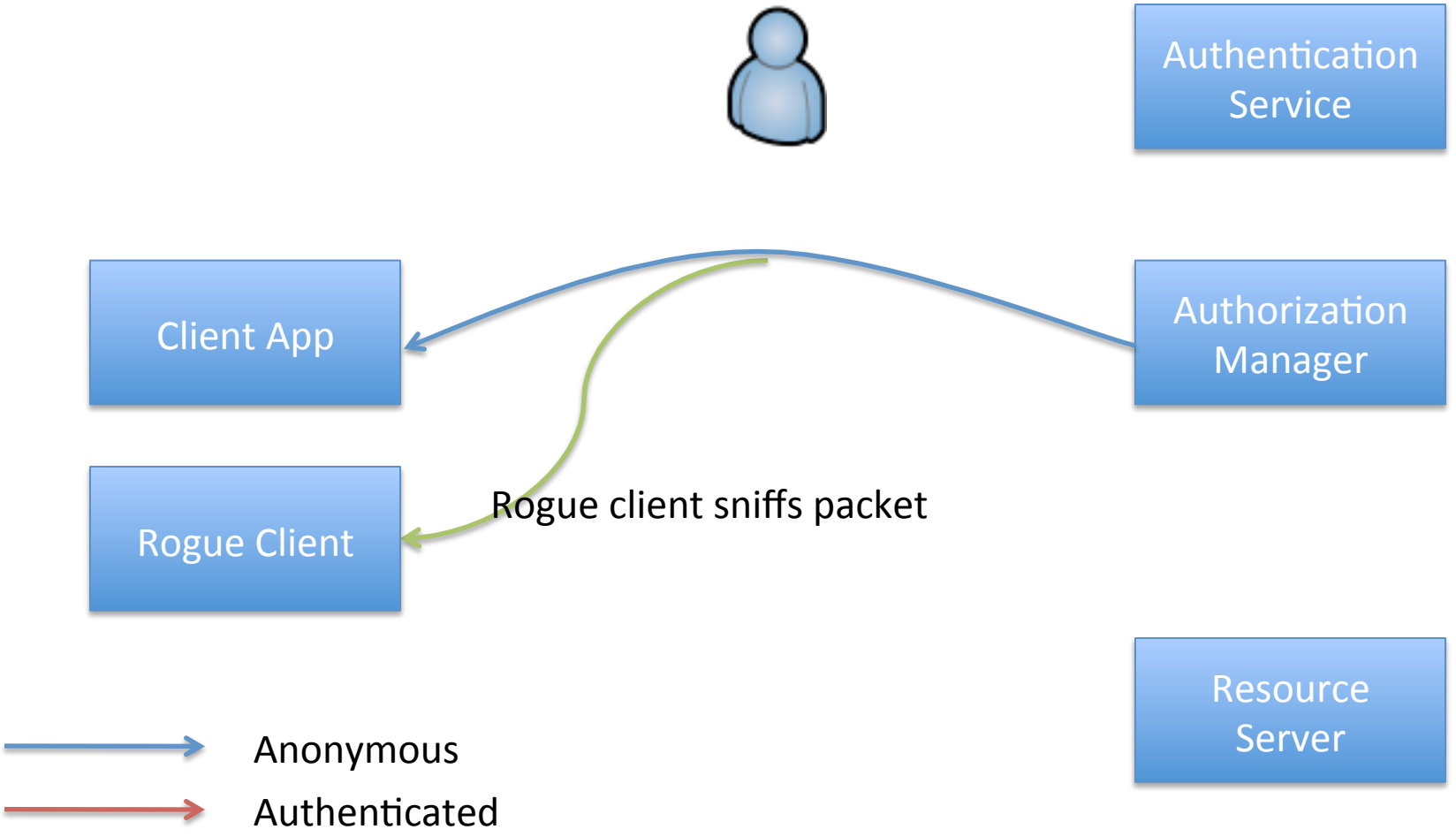
Plain

HTTPS

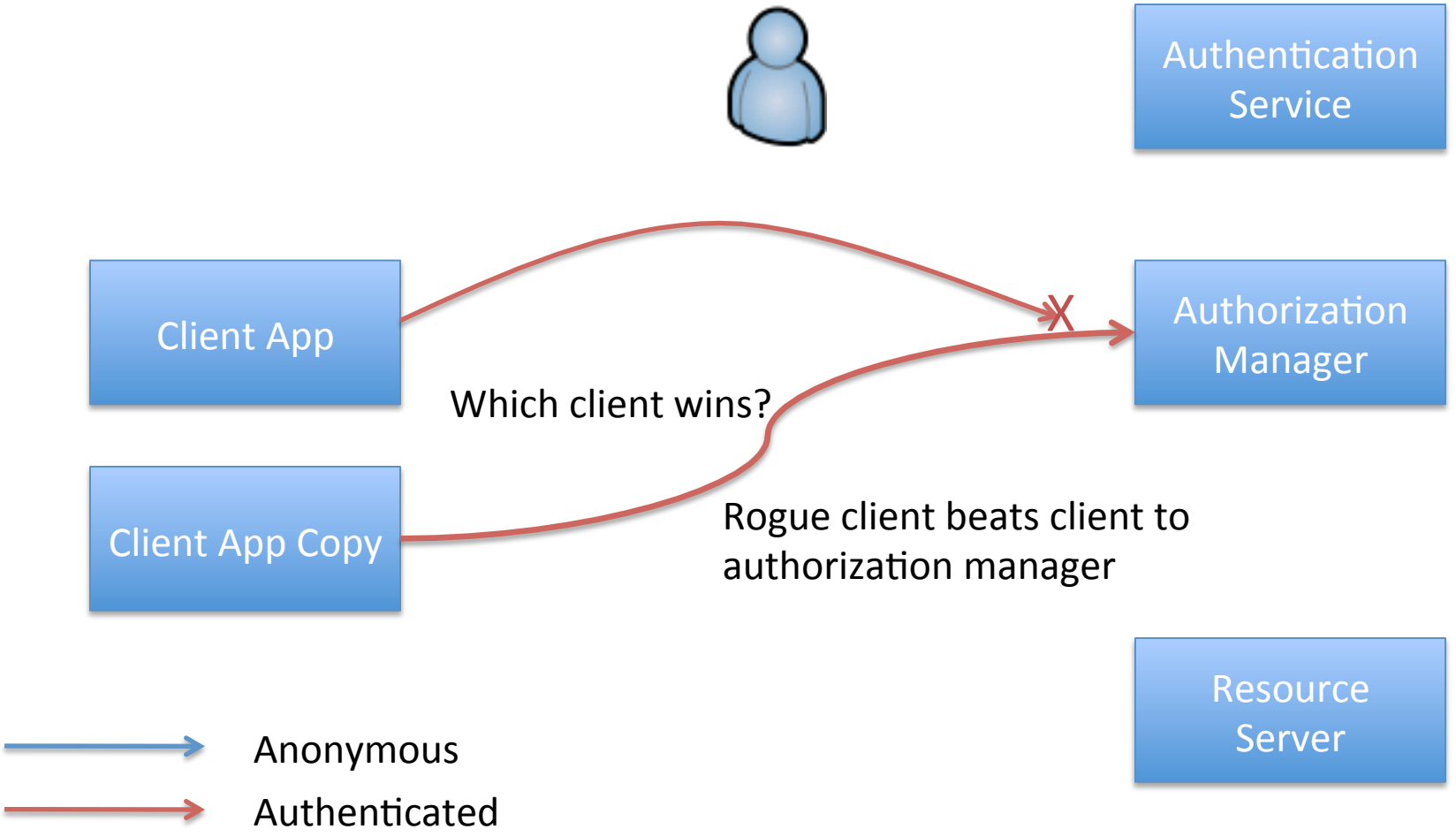
Protocol Flow



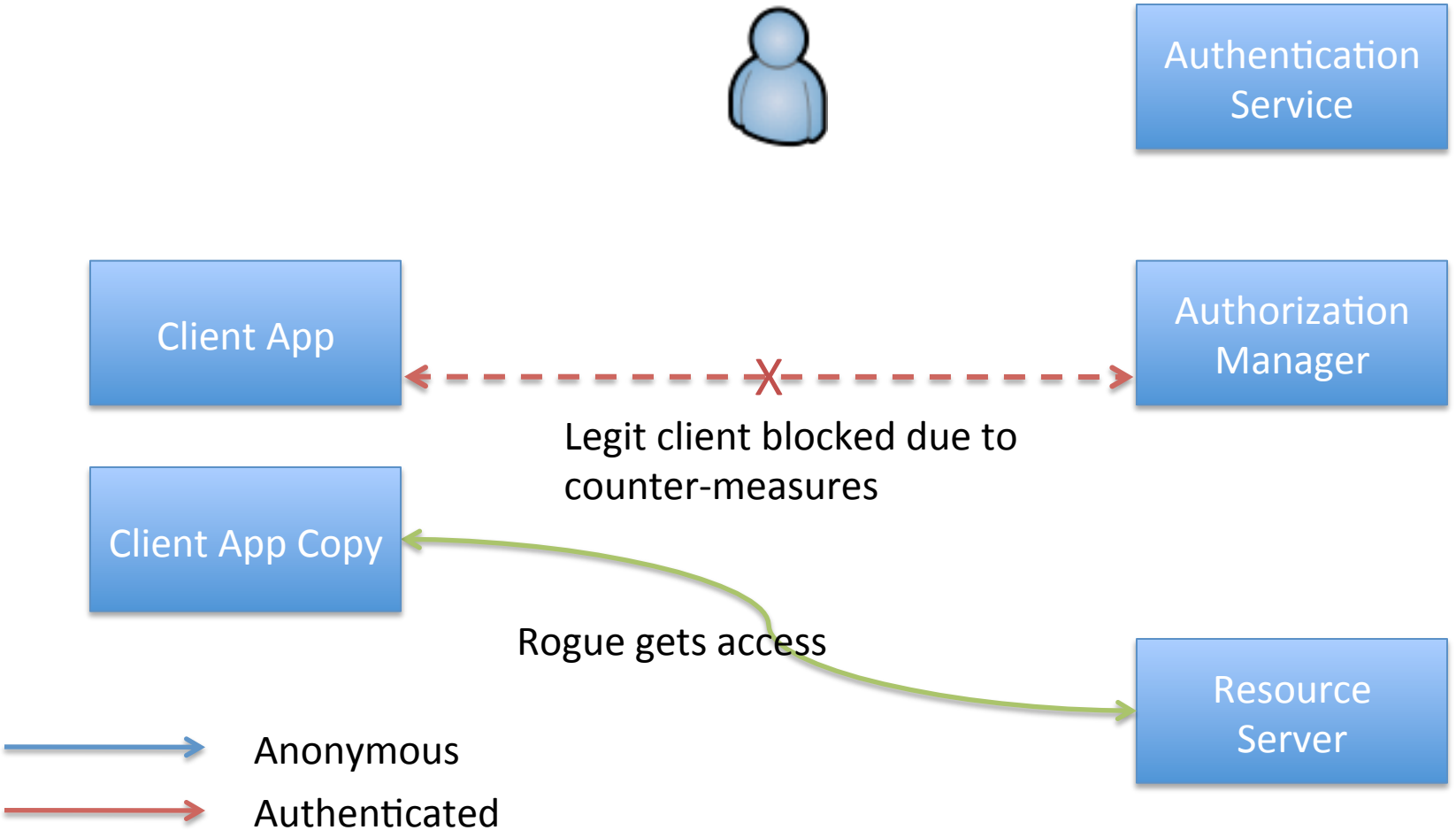
Protocol Flow



Protocol Flow



Protocol Flow



Conclusions/Discussion

- Authorization REQUEST & RESPONSE MUST be done in TLS
  - Intermediary flows SHOULD use TLS but final RESPONSE MUST use TLS
  - But: This has significant impact on some service providers!
- Alternatives: protect session authorization code
  - Bind to instance of client (client initiated secret?)
  - Make authz code non-bearer
  - Other?