Scaling Backend Authentication at Facebook

Kevin Lewi, Callen Rain, Stephen Weis, Yueting Lee, Haozhi Xiong, Benjamin Yang

Facebook


Prineville, OR


Forest City, NC


Luleä, Sweden


Altoona, IA


Fort Worth, TX


Clonee, Ireland


Los Lunas, NM

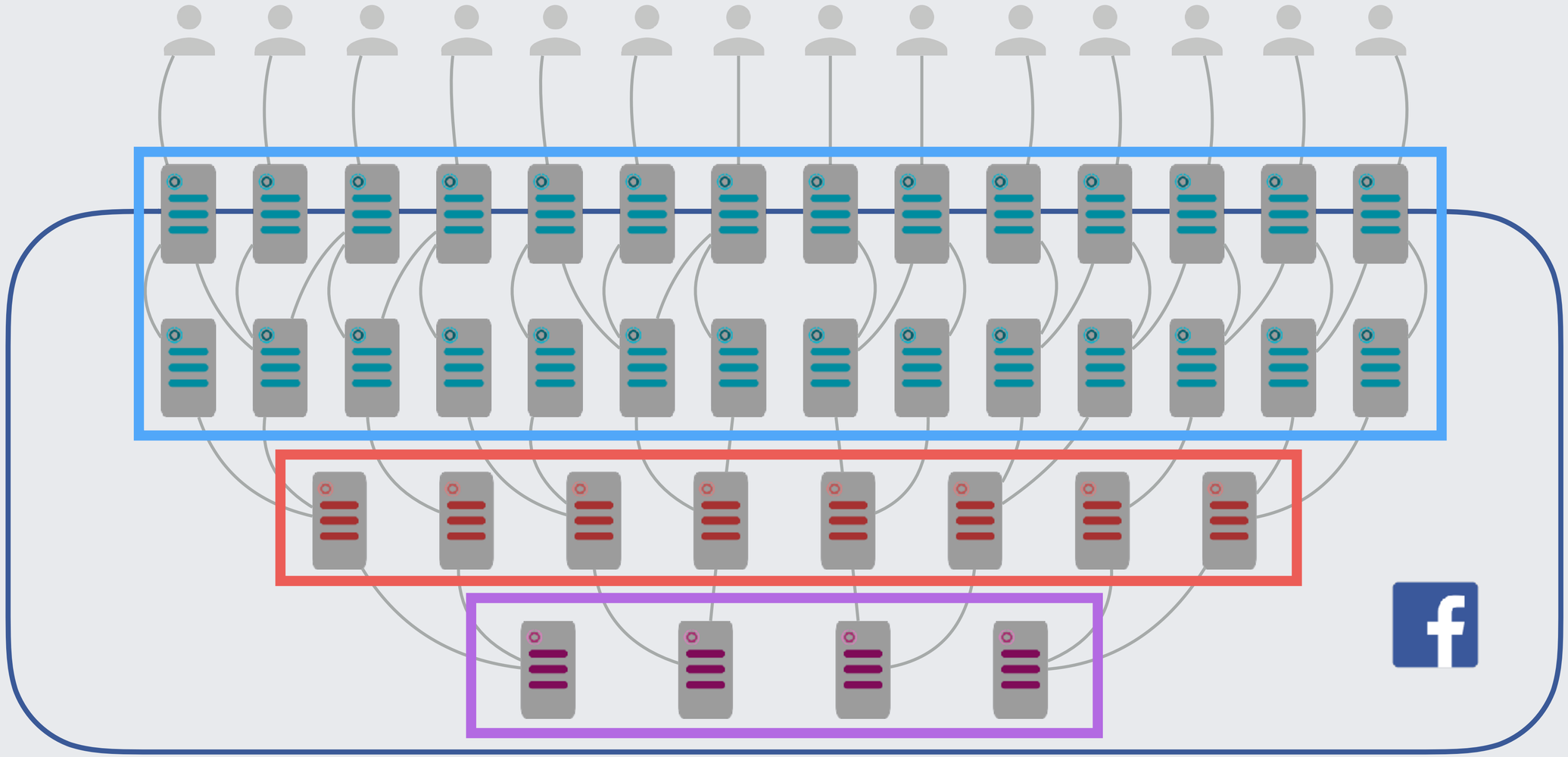

Odense, Denmark


Papillion, NE

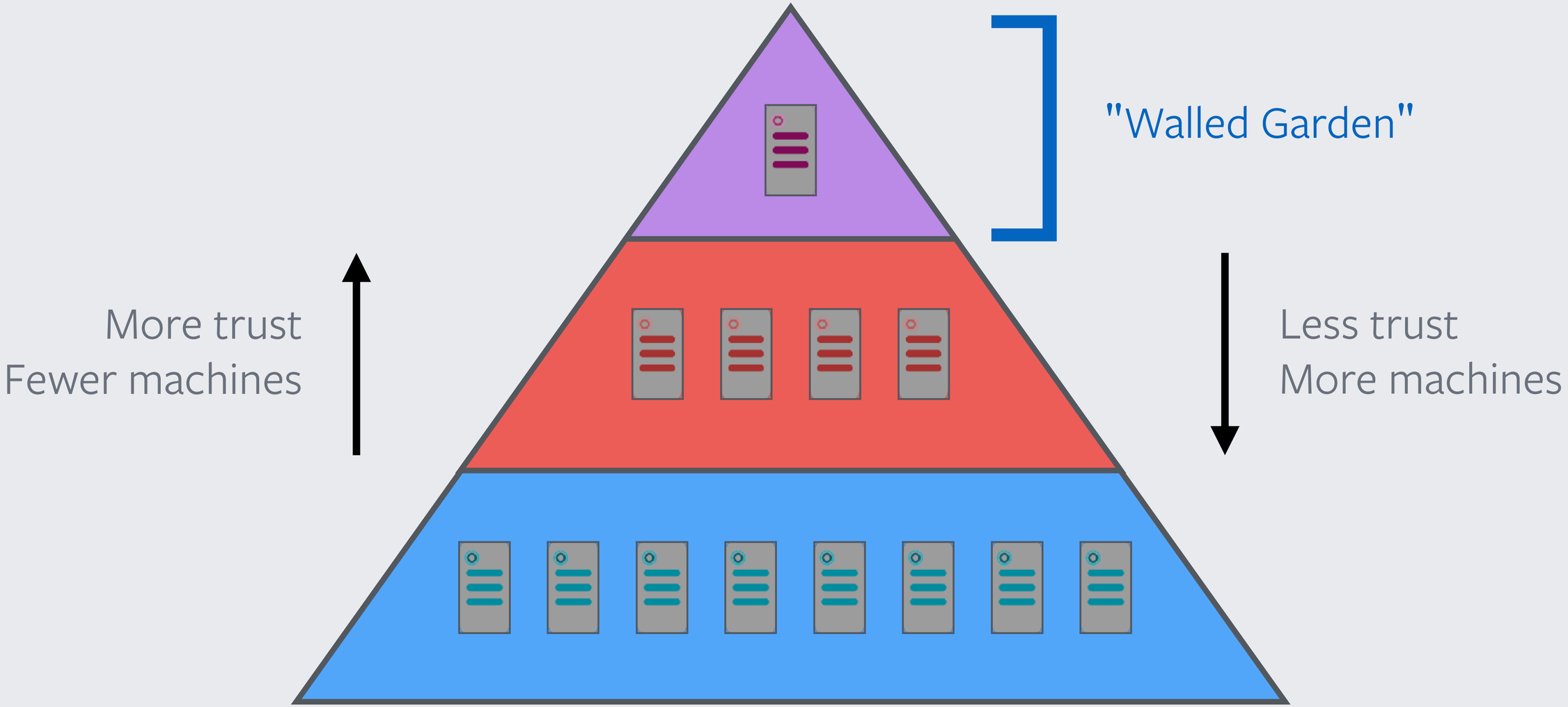

New Albany, OH


Henrico, VA

Infrastructure Security



Building from a Root of Trust



How can we scale authentication while minimizing our root of trust?

Trusted Components

Key Server
(Holds Master Keys)

Root CA
(Signs Certificates)

Login Server
(Signs Sessions)

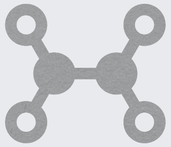
Authorization Server
(Signs ACLs)

Authentication and Authorization

Identities

User: ″Callen Rain″

Machine: server123.fb.com



Service: Image Uploading

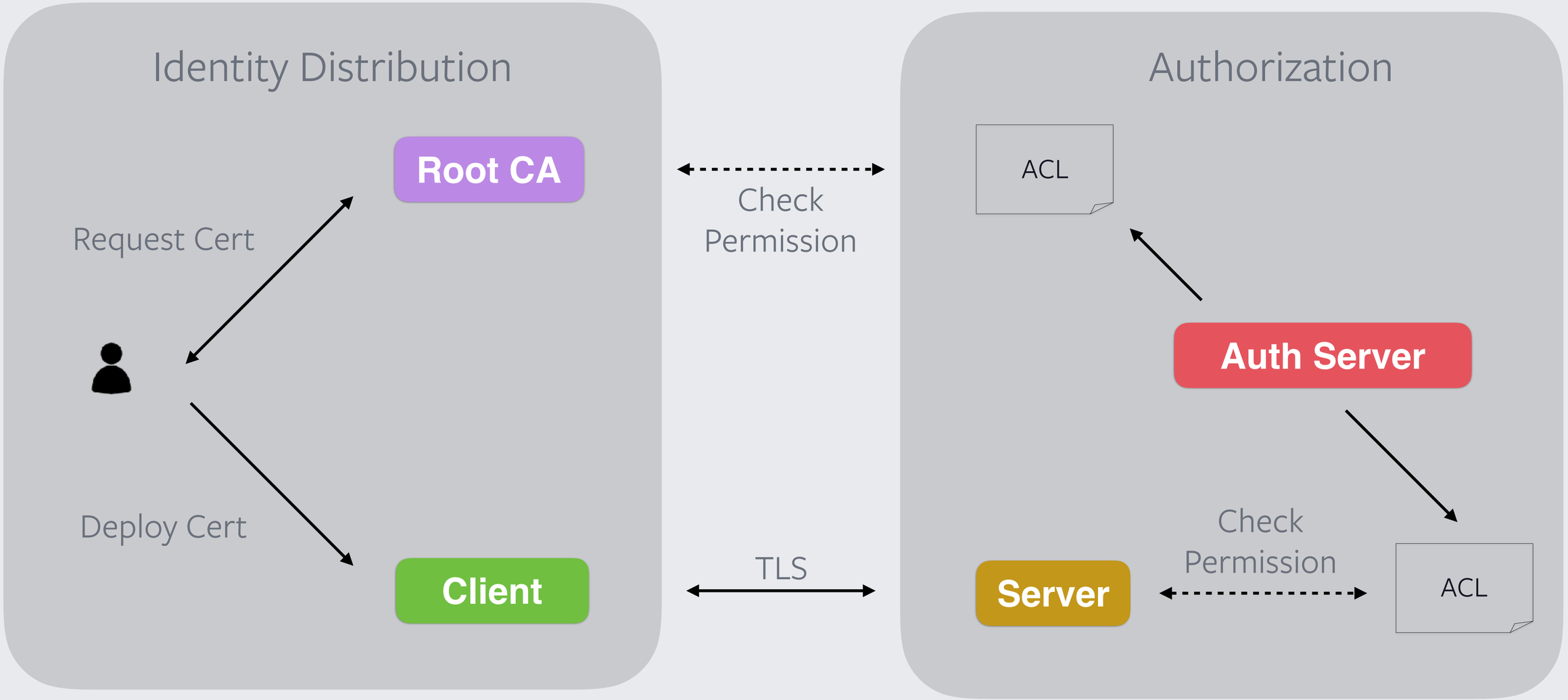
Access Control Lists (ACLs)

Resource:
″Who can access table X in database Y?″

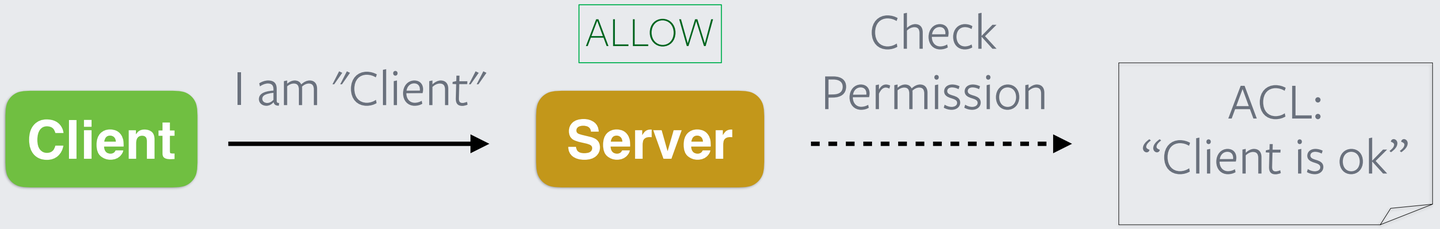
- Identity1
- Identity2

...

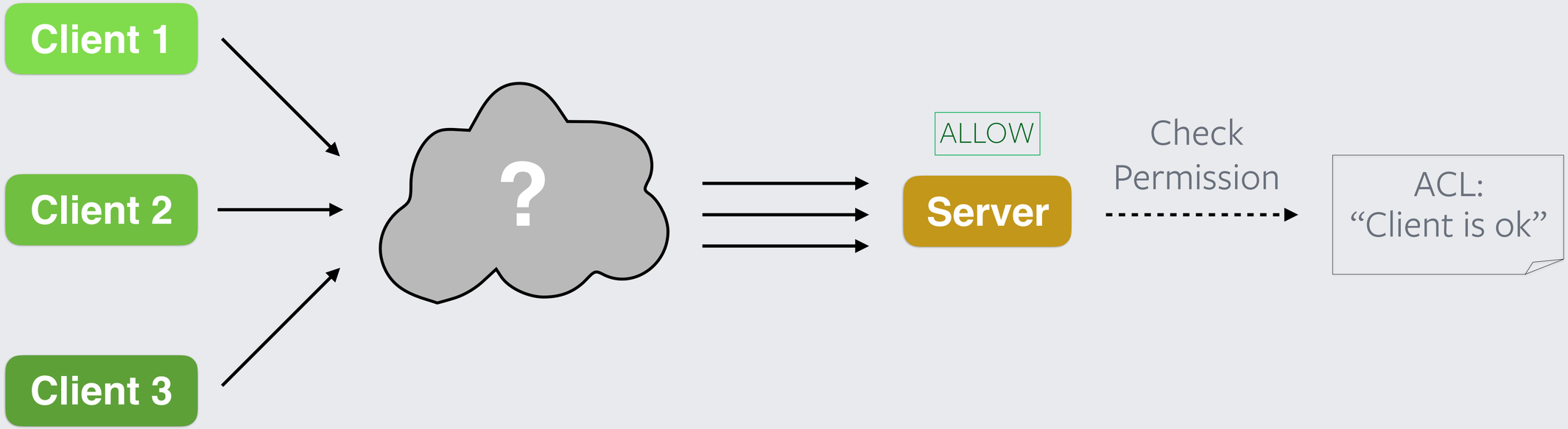
Service Authentication with TLS



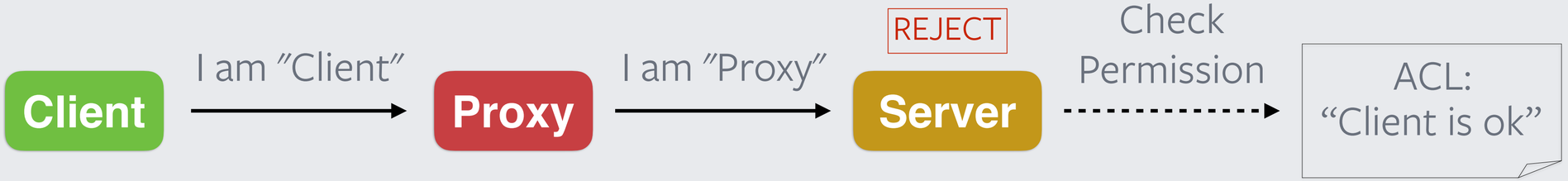
Service Authentication with TLS



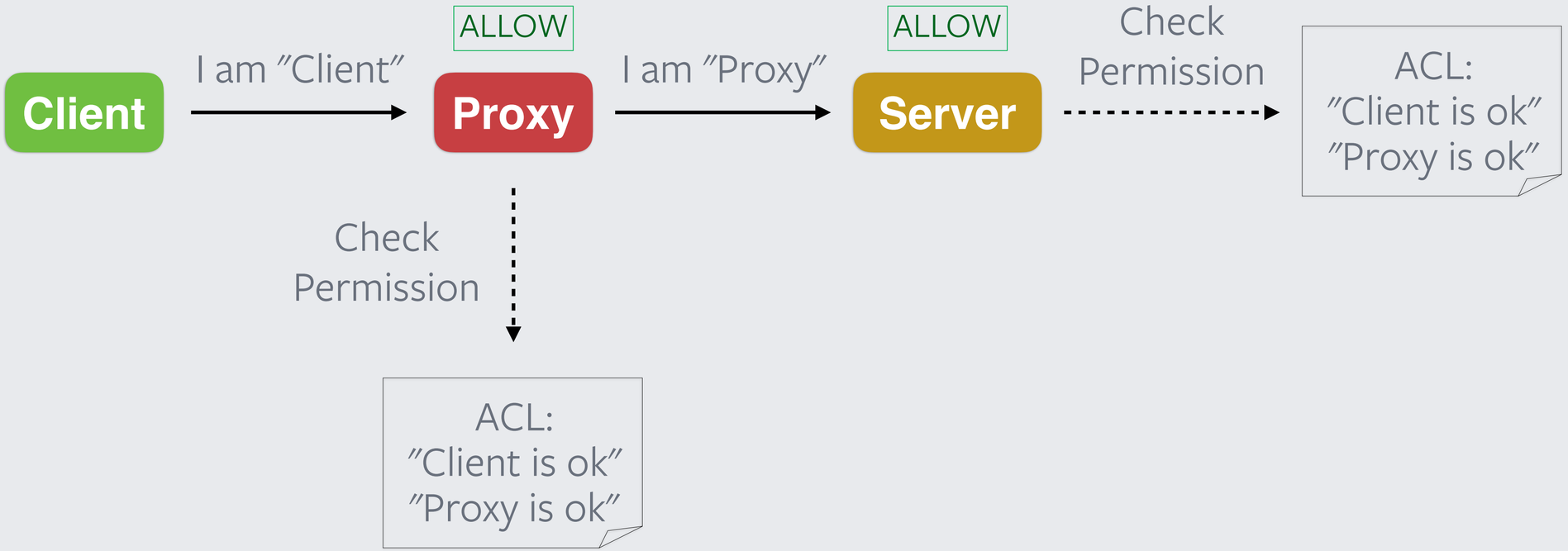
Service Authentication with TLS



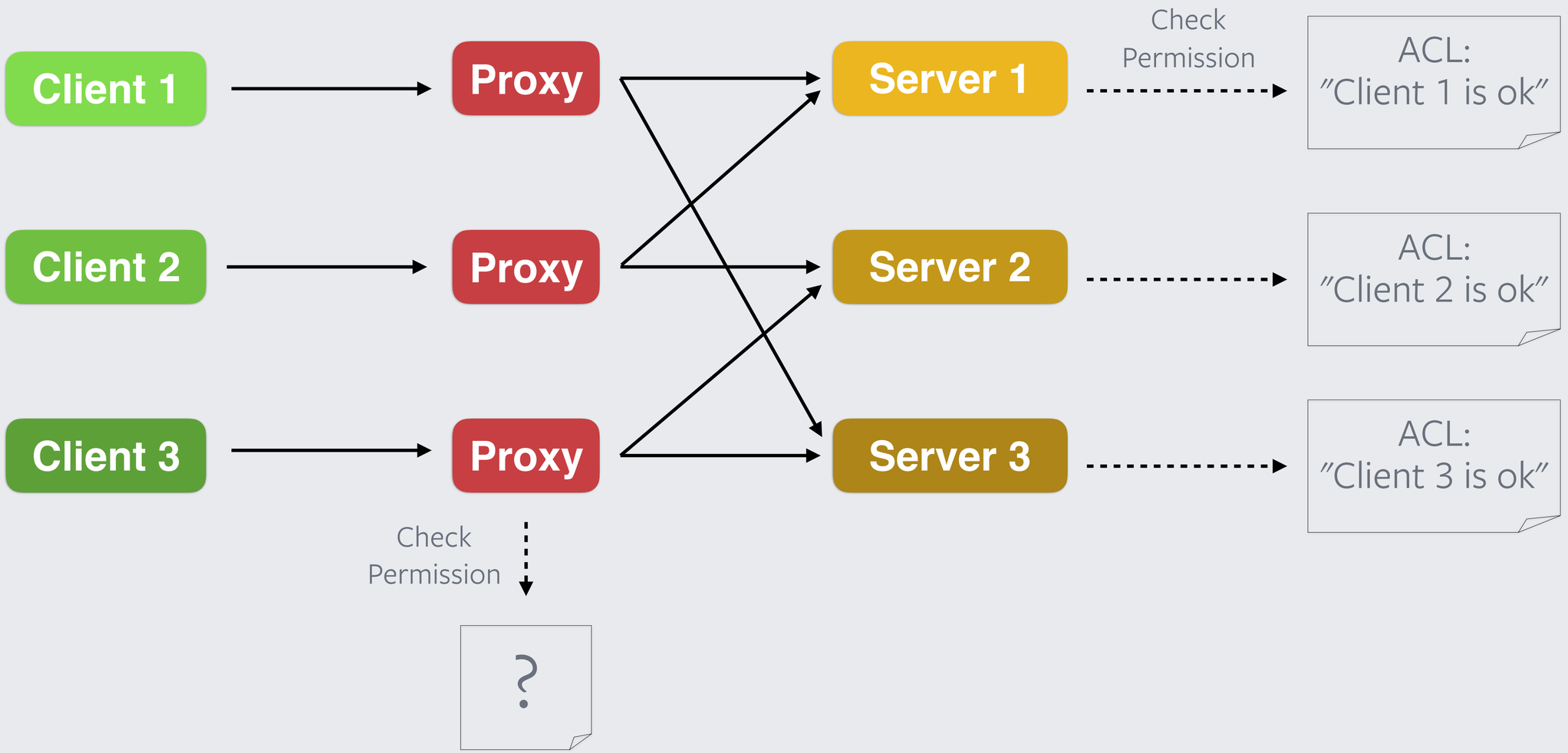
Intermediate Proxies



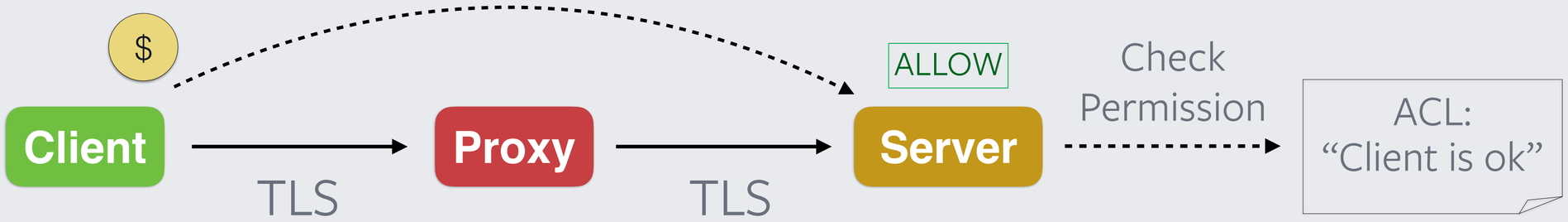
Intermediate Proxies



Intermediate Proxies



Tokens

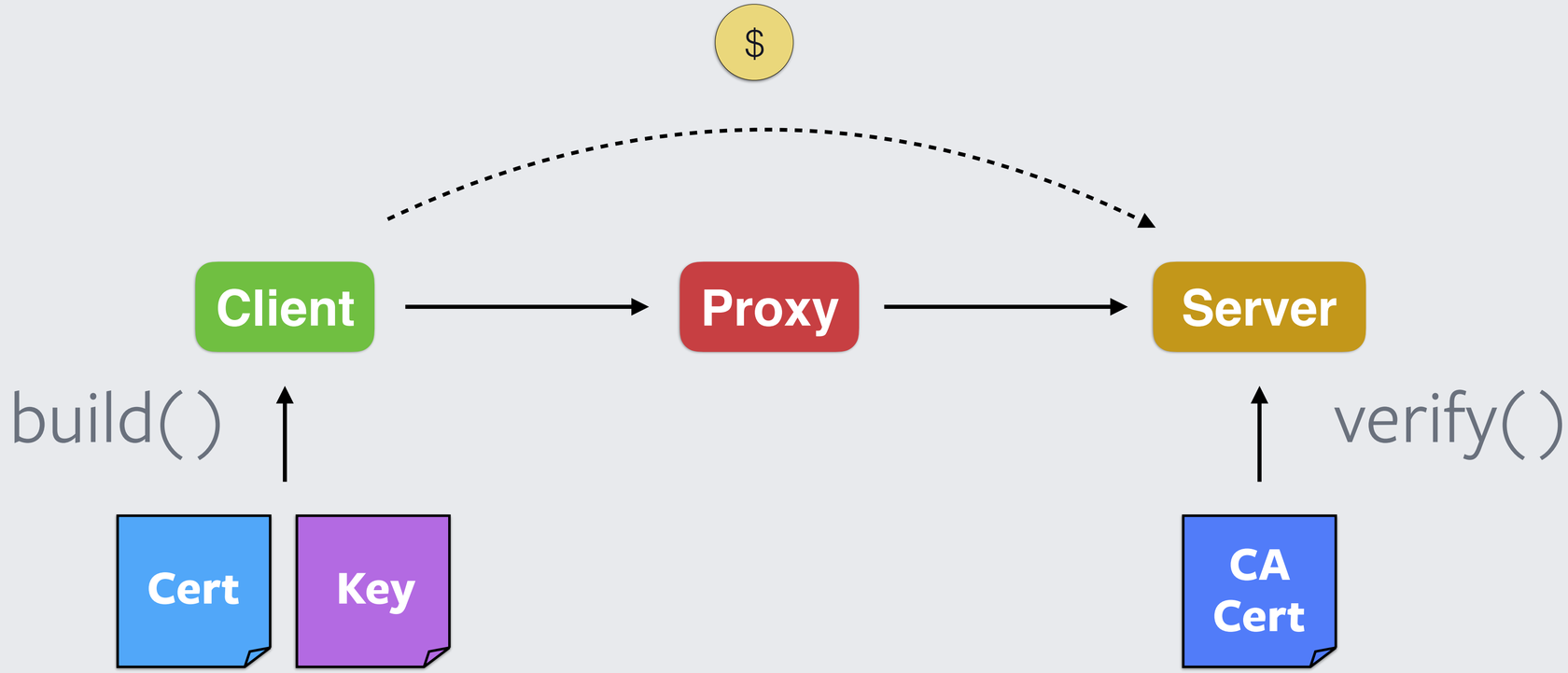

$
Client
TLS
Proxy
TLS
ALLOW
Server
Check
Permission
ACL:
“Client is ok”

Tokens

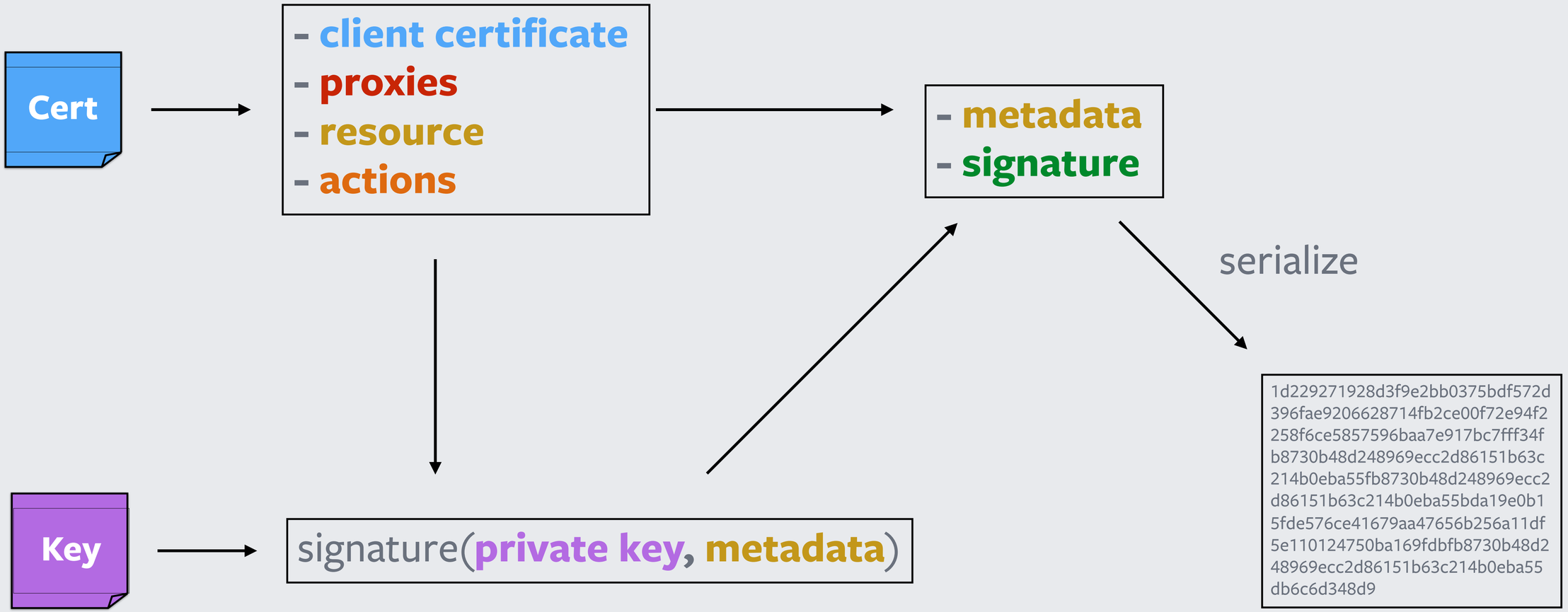
1. Certificate-Based Tokens

2. Crypto Auth Tokens (CATs)

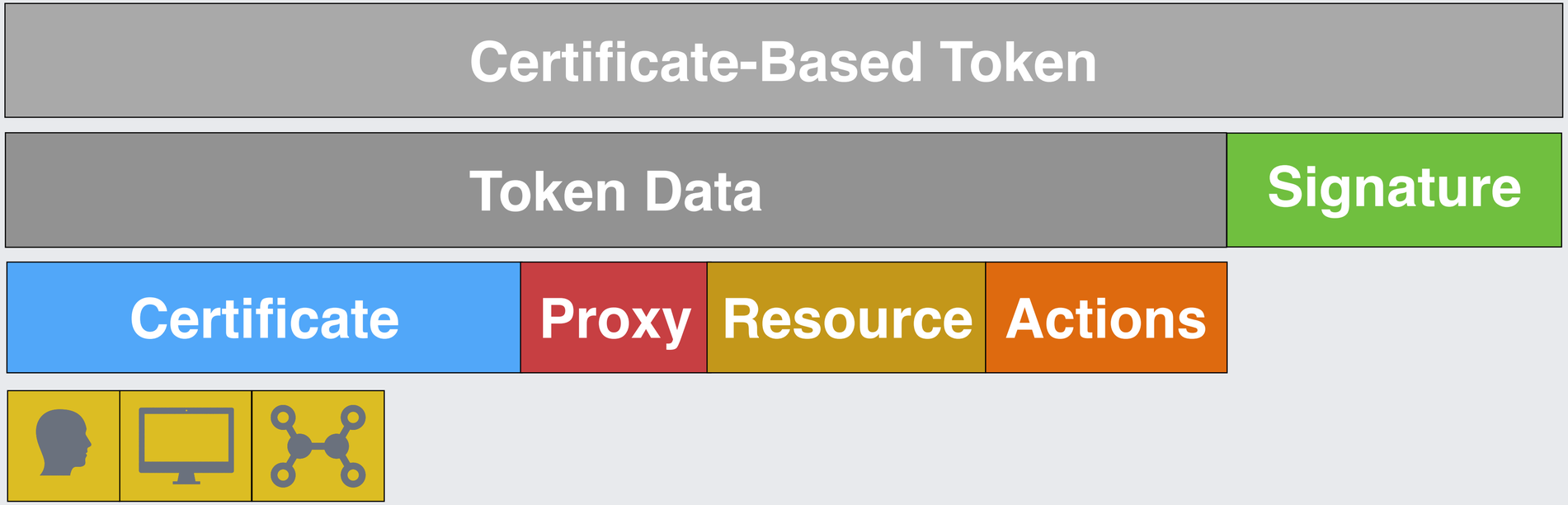
Certificate-Based Tokens



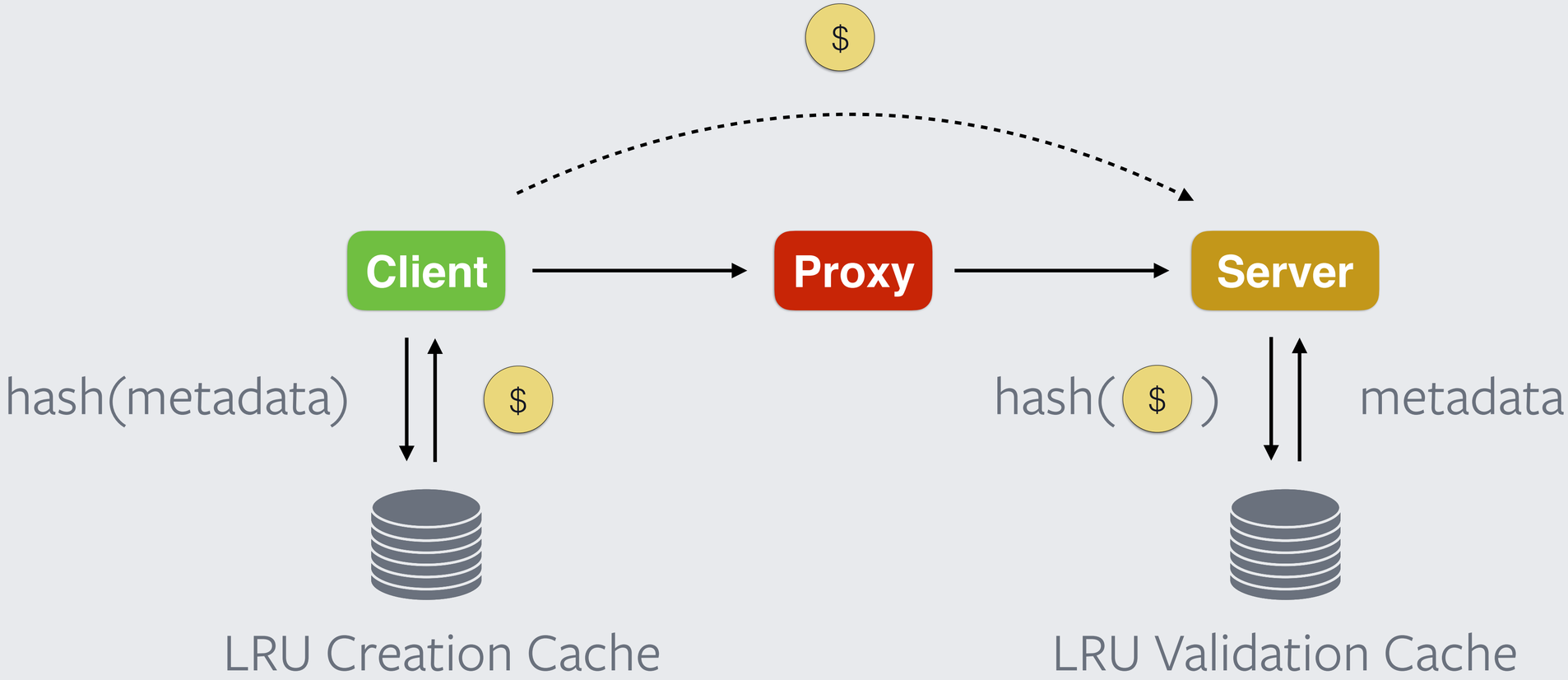
Certificate-Based Token Creation



Certificate-Based Token Verification



Caching Certificate-Based Tokens



Tradeoffs with Cert-Based Tokens

Pros

Reliable

Simple

Generic

Cons

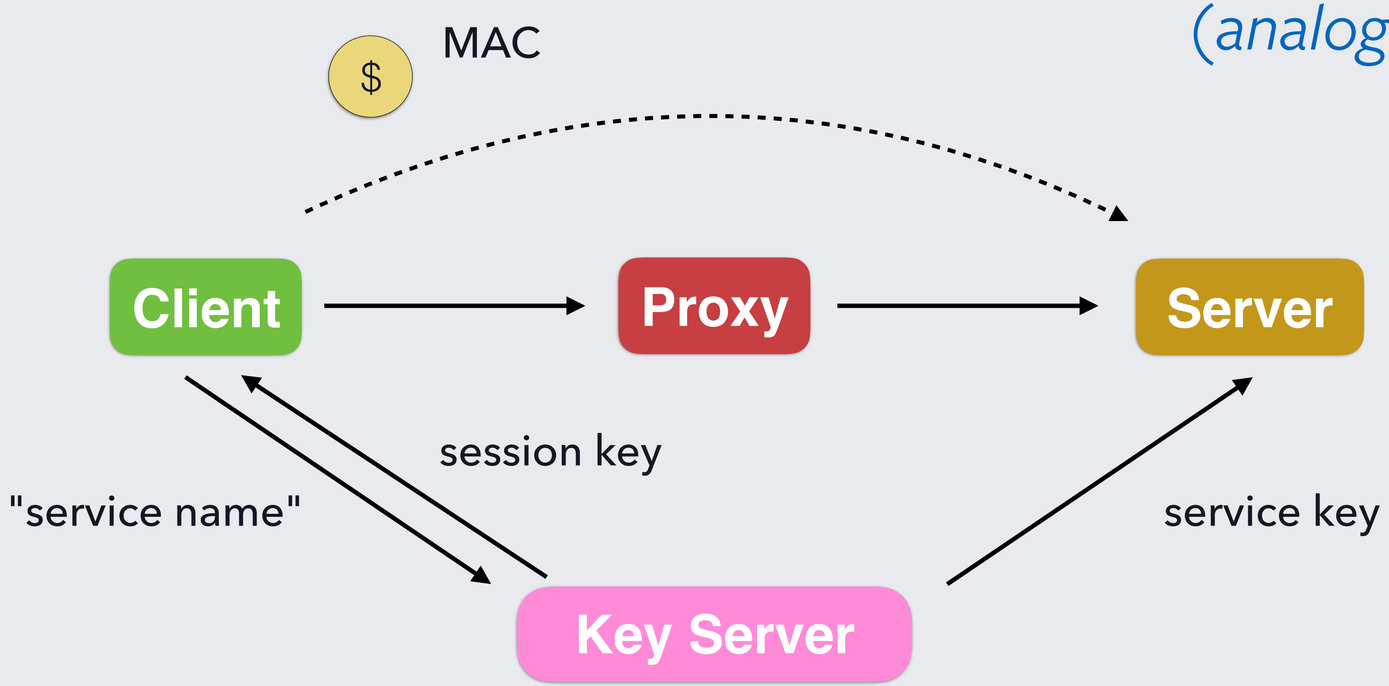
Large

Public-Key

x509

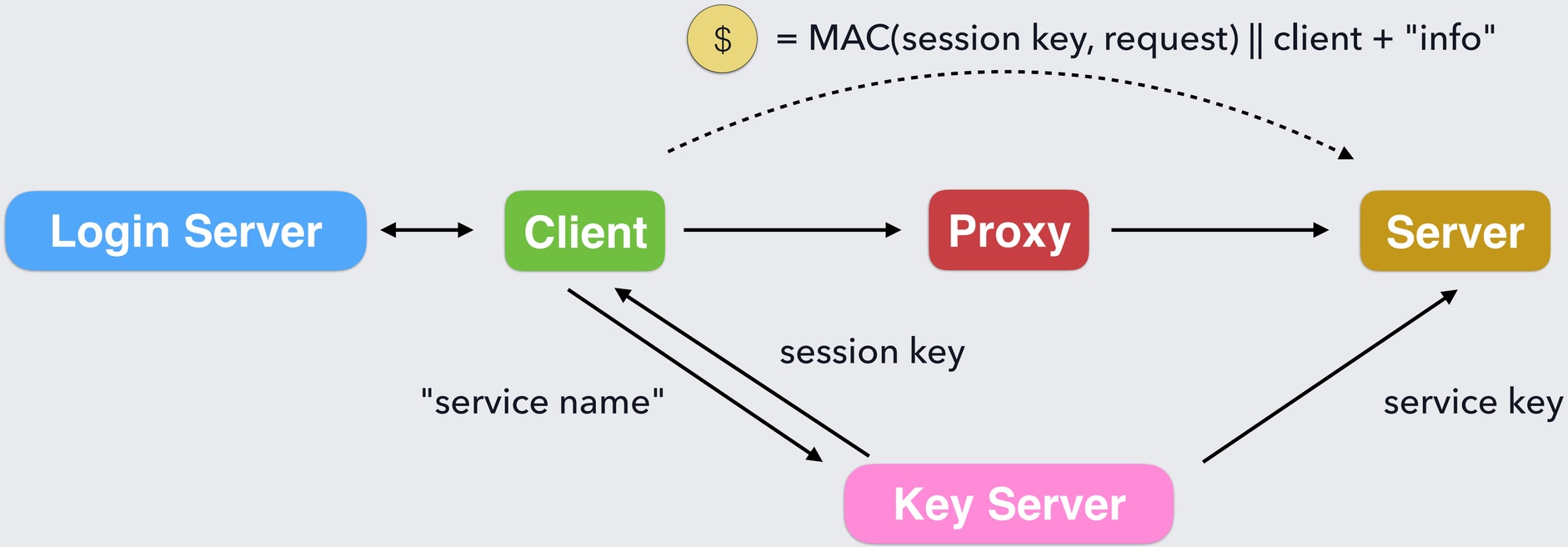
A Symmetric-Key Variant

(analogous to Kerberos)



All direct communications are encrypted / authenticated with TLS

”Crypto Auth Tokens” (CATs)



service key = PRF(master key, "service" + info)

session key = PRF(service key, "client" + info)

All direct communications are encrypted / authenticated with TLS

Summary

1. We build from a small root of trust

2. TLS by itself isn't enough

3. Tokens
    - Public-Key
    - Symmetric-Key

Acknowledgments