Deploying QUIC at Scale
(at Google)

Ian Swett, Martin Duke

Agenda

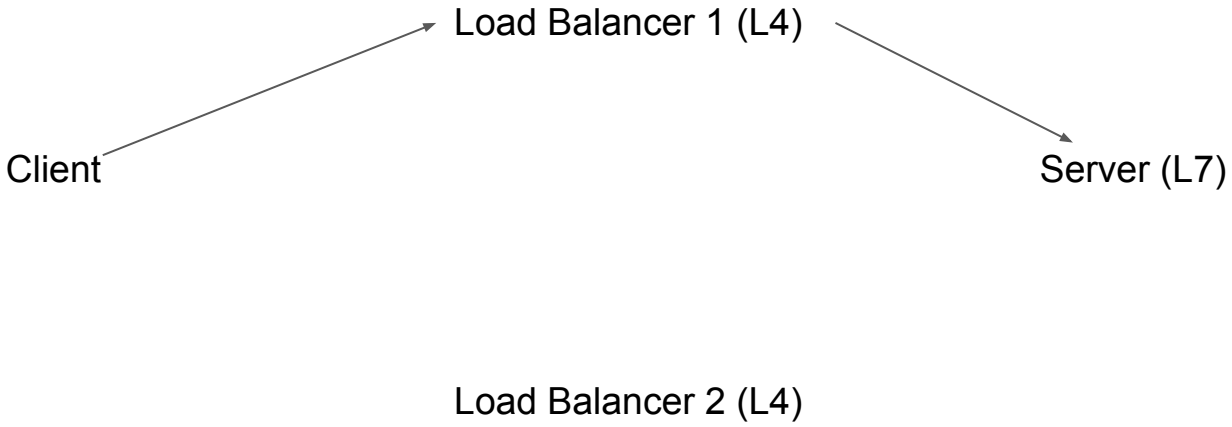
1. QUIC Load Balancing
2. QUIC blackholing
3. A QUIC outage
4. 0-RTT in IETF QUIC

Load Balancing

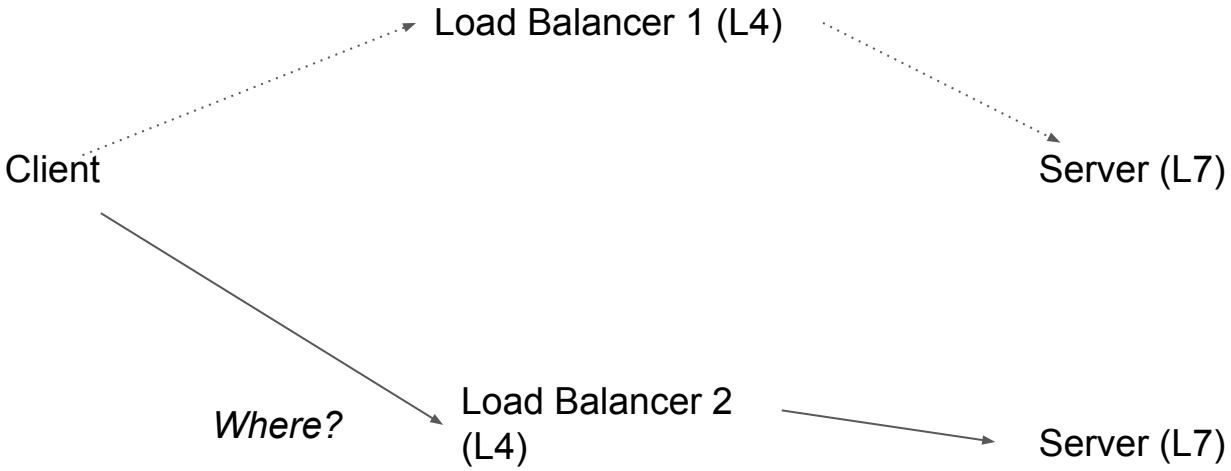
A QUIC plug for QUIC-LB

Anycast

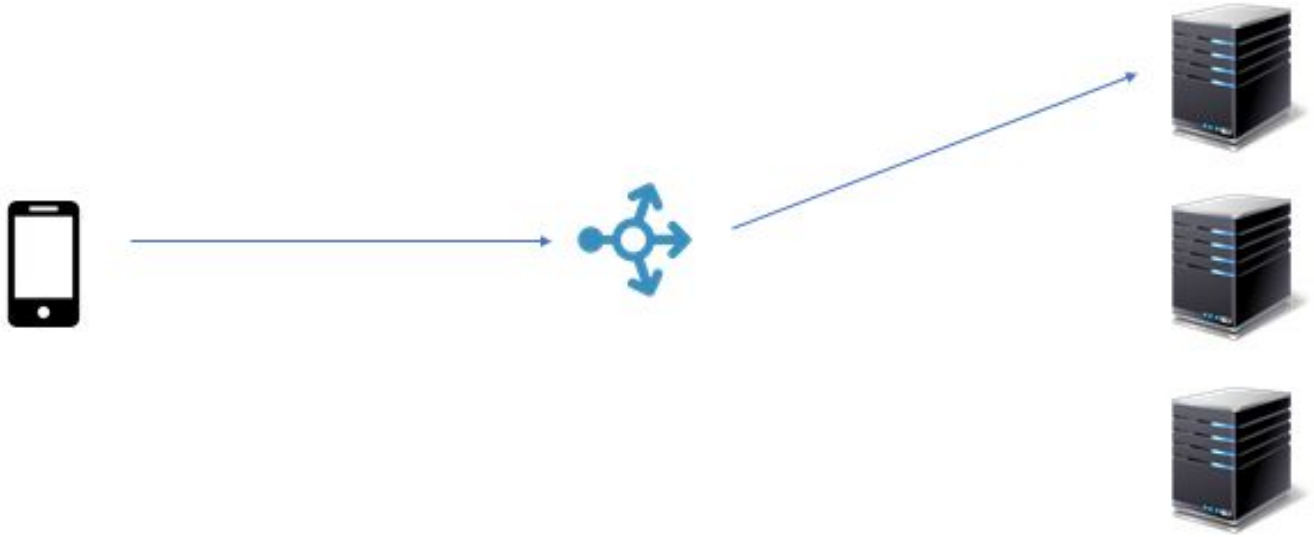
A single IP shared using BGP for load balancing



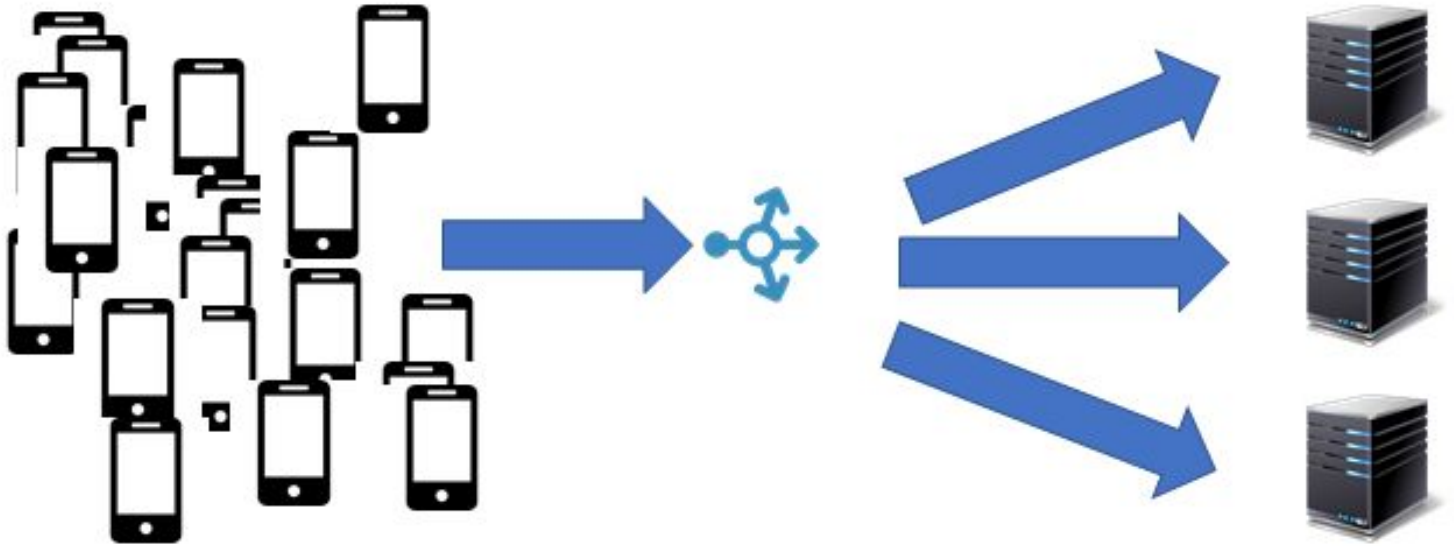
Anycast


Load Balancer 1 (L4)
Client
Server (L7)
Where?
Load Balancer 2
(L4)
Server (L7)

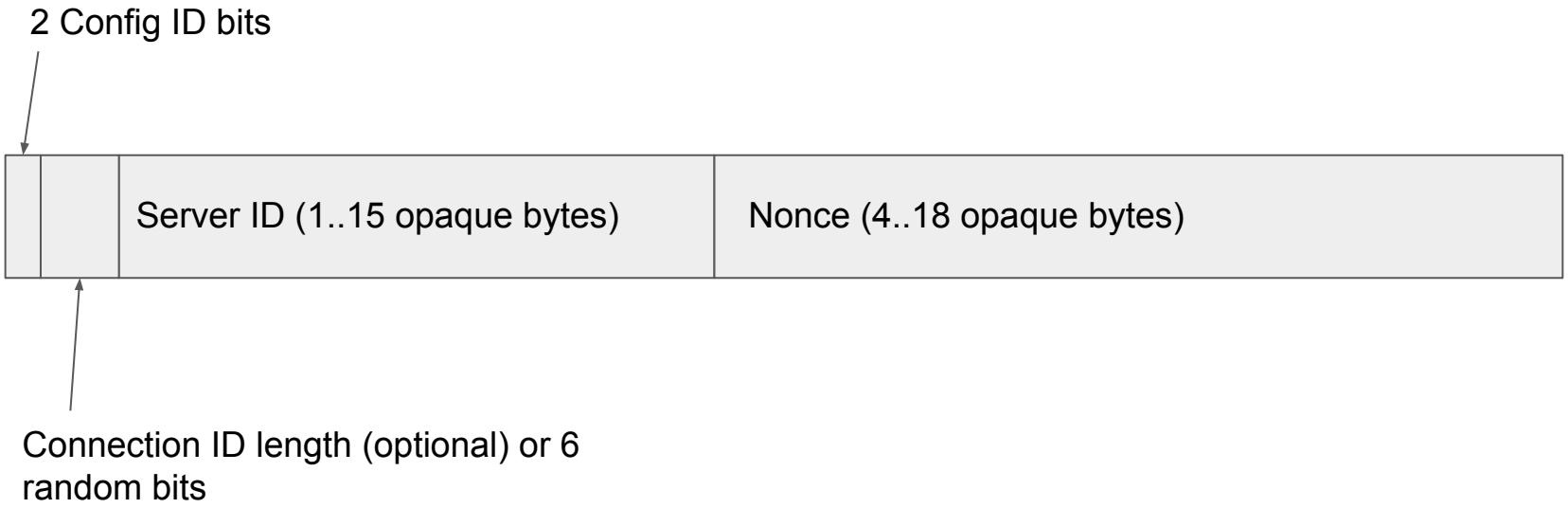
Perfect Linkability



Perfect Unlinkability



One Connection ID format



Config id ‘11’ = unroutable

Could be encrypted or not – algorithm depends on length

QUIC Blackholing

(post-handshake)

5-tuple blackholing

A 5-tuple can be blackholed, even if most 5-tuples between two endpoints work

- Maybe it traverses a broken piece of hardware?
- Maybe a machine has a bad line card?
- Maybe the internet is a terrible broken place?

Blackholing can cause QUIC to wait for idle timeout, 30s-minutes

What we've done to mitigate it

To reduce the time to connection failure,
close the connection after consecutive (5) PTOs

Reduces tail latency substantially

Probably closes a few 'good' connections, unfortunately

Requests still fail, but many can be retried by the browser or app

We do this on the server or client,
though it's unclear why it helps so much on the server side?

A QUIC Solution

Observation: Changing only port can drastically change the path

ie: entirely different datacenters or peering points.

Try a new client ephemeral port!

Introduces entropy in both directions, direction doesn't matter

No need for privileged access

Default enabled in Chromium (ie: Chrome, Cronet, …)

QUIC Exit and Contagion Bugs

A short summary of FB Reliability@Scale (Recording, Slides)

Summary

Query of death triggered by resumption information sent *from GFEs to clients and back to GFEs* caused GFEs to crash.

At peak around **10% of Google GFEs** were crashing, but this distribution was very uneven.

Impact was mostly limited to Europe, and to services served from datacenters.

Total outage time was **1h 44m**.

Contagion: An interaction of distributed systems

Slow rollouts identify most bugs before significant harm

- If a bug is found, roll back.

Contagion bugs are **not** fixed by rollbacks alone.

A single task could cause a global outage.

- Persistent state in another system is not rolled back.
- In the case of internet clients, cannot rollback.

Example: TLS or QUIC Resumption

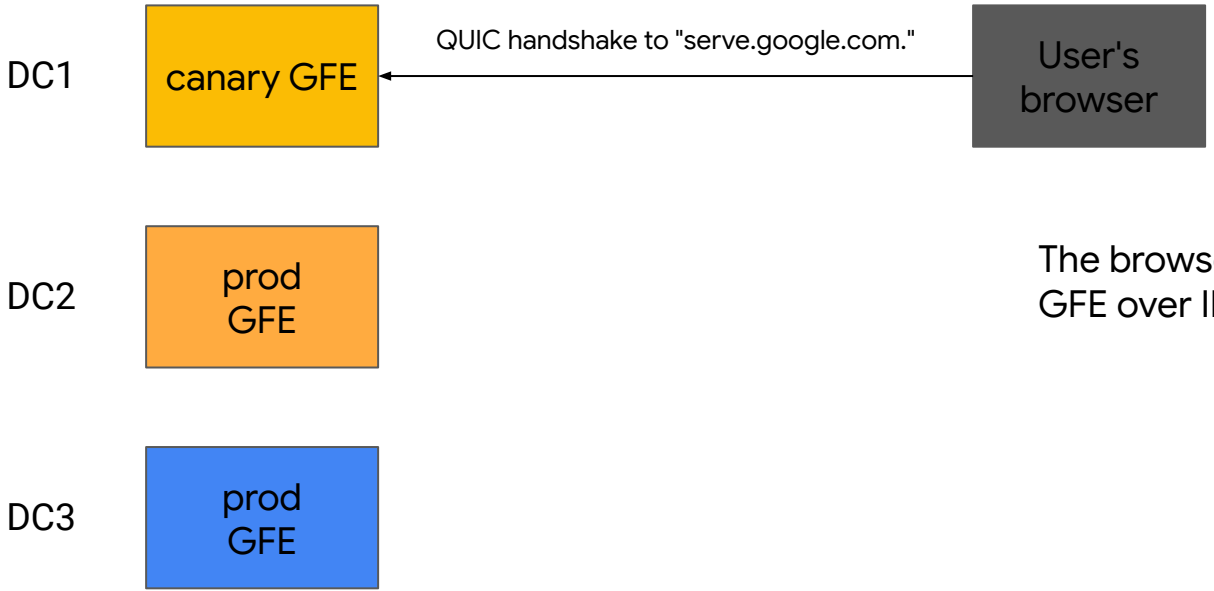
- TLS resumption
- QUIC source address tokens
- gQUIC server configs

One GFE gives the client information for a future connection

Another GFE parses it later and something goes wrong.

What happened at Google in November 2021

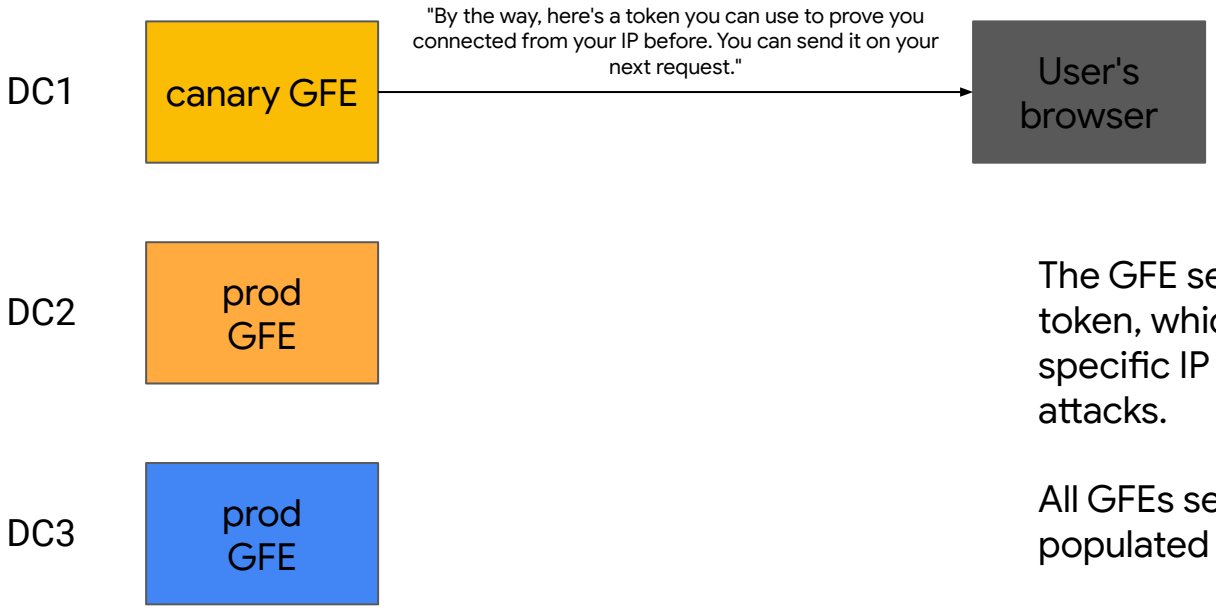
Mechanism of action



The browser completes a handshake to a GFE over IETF QUIC.

Source: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis non erat sem

Mechanism of action

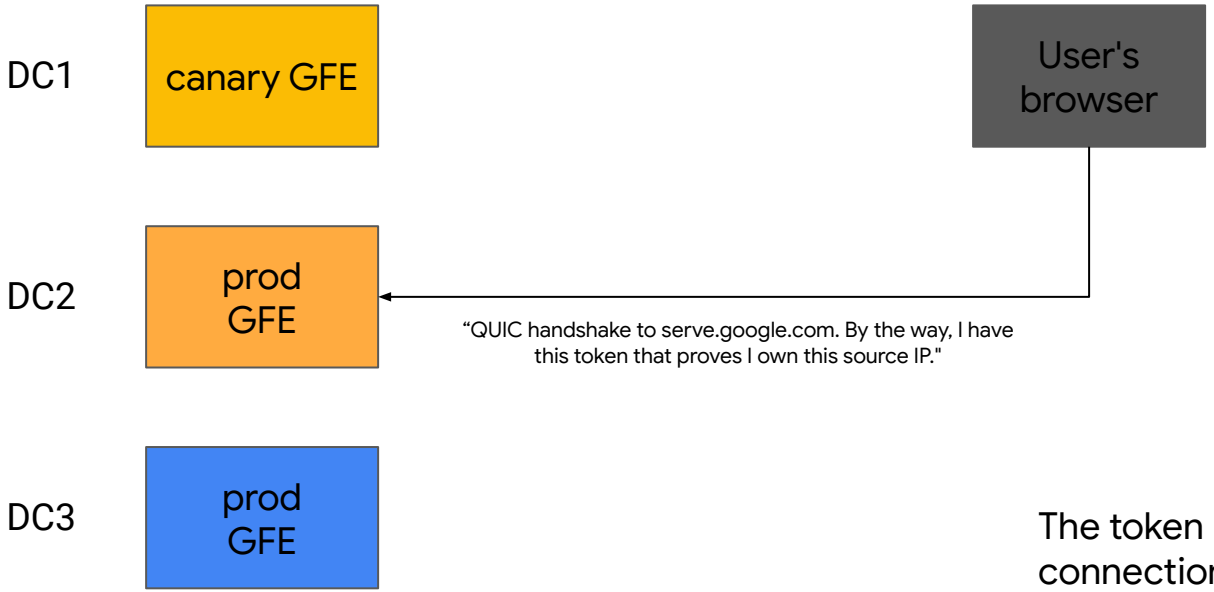


The GFE sends the browser an encrypted token, which proves a client owns a specific IP address, limiting amplification attacks.

All GFEs send this token, but canary jobs populated a new field.

Source: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis non erat sem

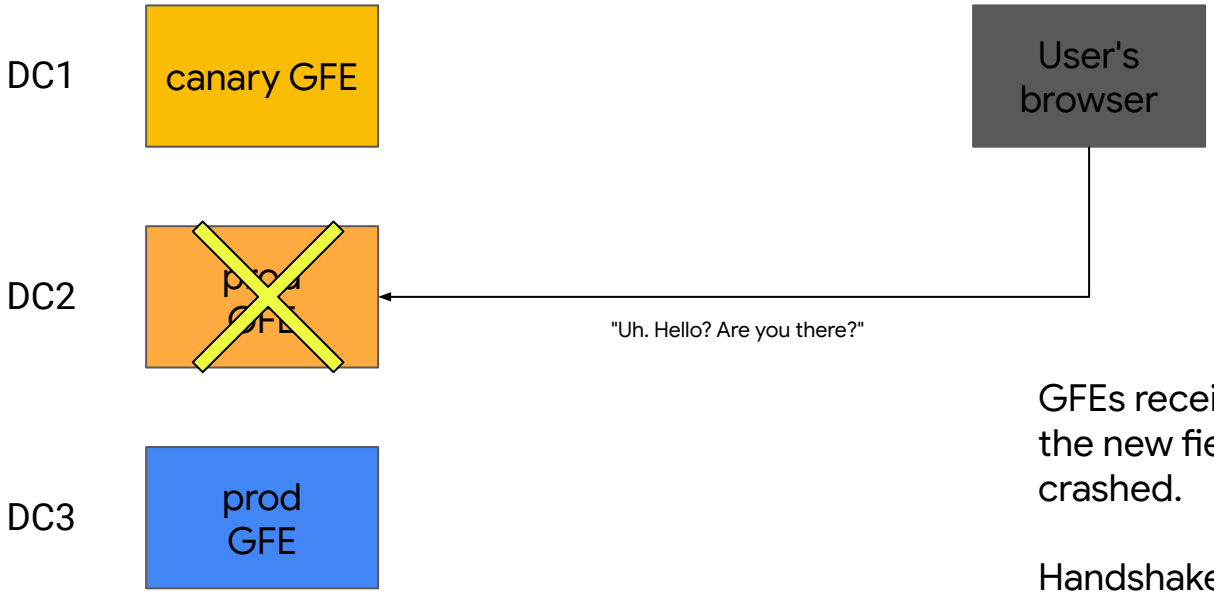
Mechanism of action



The token is sent by the client on the next connection; after a handshake, the token *should* be cleared and was if the handshake completed.

Source: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis non erat sem

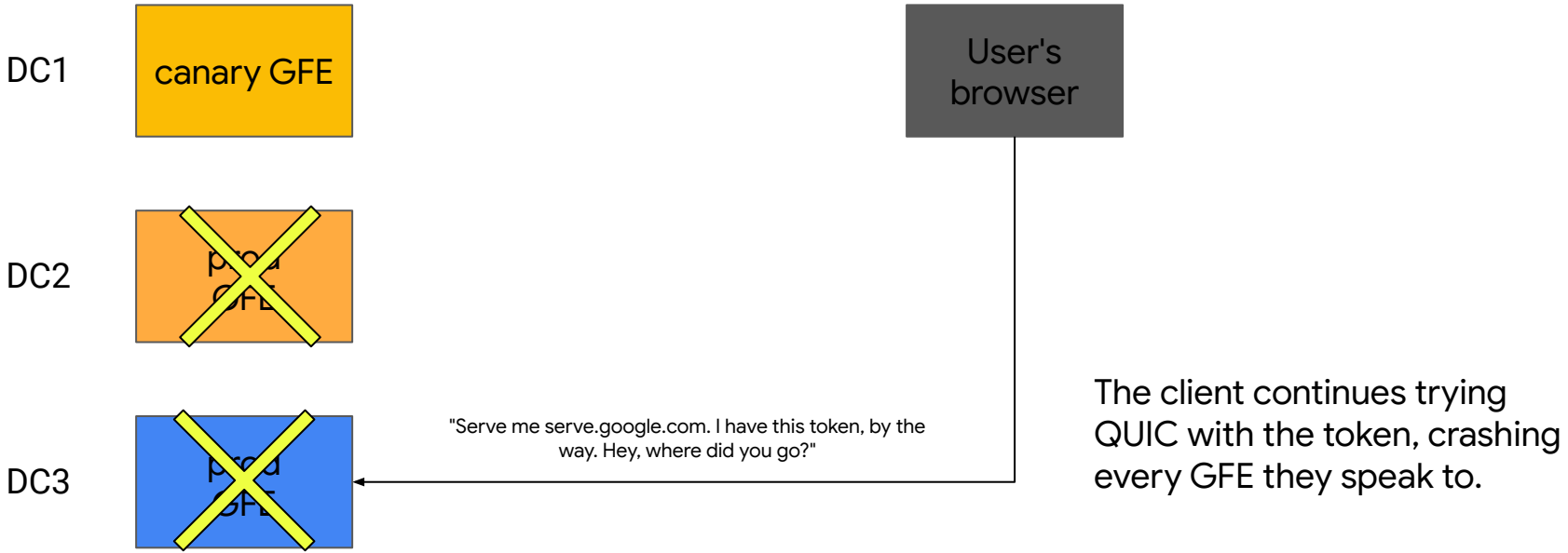
Mechanism of action



GFEs receive the IETF QUIC token with the new field, dereferenced a nullptr, and crashed.

Handshake doesn't complete, so due to a bug, the client keeps using the "poison" token.

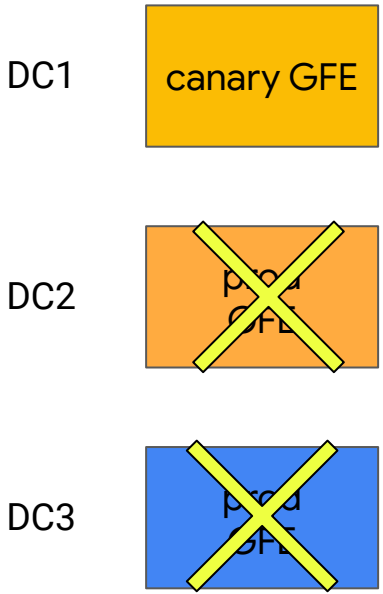
Mechanism of action

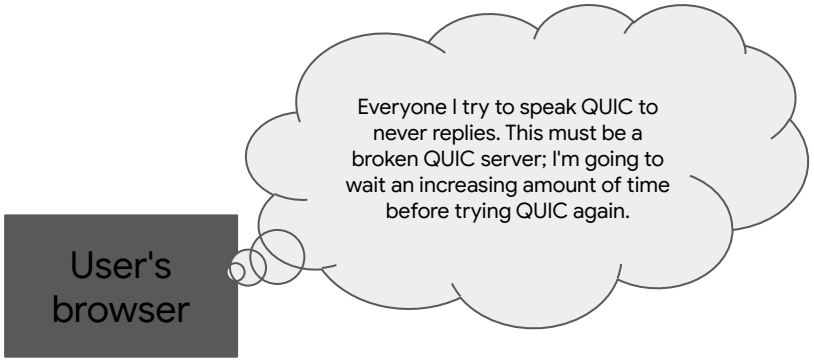


The client continues trying QUIC with the token, crashing every GFE they speak to.

Source: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis non erat sem

Mechanism of action





When Chromium clients see a handshake failure, they mark QUIC is "broken", and go into exponential backoff. 5 minutes... 10 minutes...

Source: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis non erat sem

From canary to resolution

00:27 PST

4 Canary GFEs receive updated flags

GFEs in Europe begin crashing

00:31 PST

Probers fail and SREs alerted by pages

Canary judge automatically rolls back flags after 4 minutes

00:42 PST

European GFEs continue crashing

London SREs realize all monitoring tools, including crashlogs, are inaccessible

01:38 PST

London SREs learn it's not a global outage

India, NZ SREs reroute all European UberProxy traffic

01:51 PST

SREs disable QUIC

Page me at 6am to figure out what happened

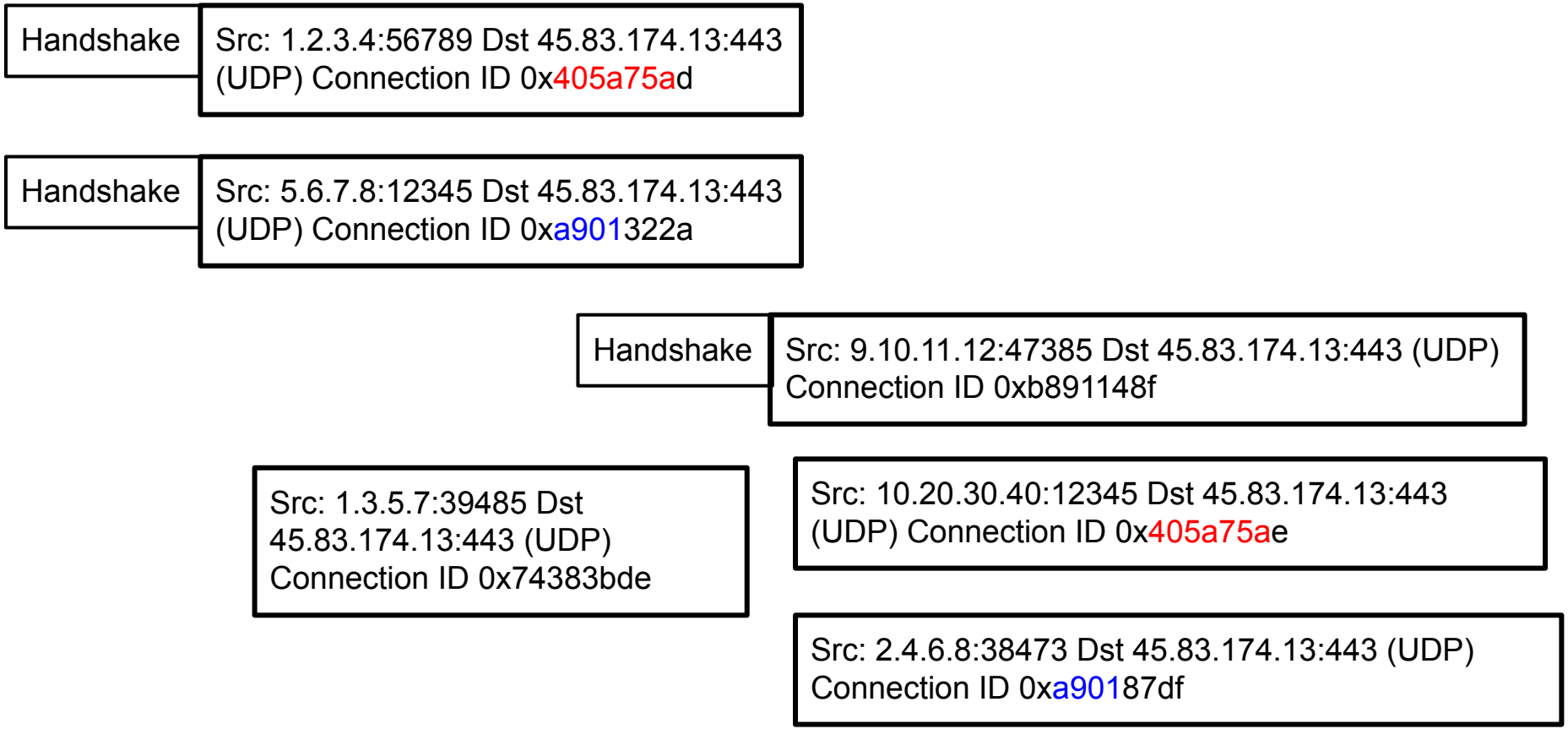
Challenges of 0-RTT

0-RTT is hard, much harder in IETF QUIC than gQUIC

- IETF QUIC can perform better that gQUIC… after fixing many bugs
- Packet Number Spaces add complexity, particularly in combination with PTO
- Key management is less synchronous than TLS over TCP

Facebook Networking@Scale talk: Recording, Slides

Thanks!


Handshake
Src: 1.2.3.4:56789 Dst 45.83.174.13:443
(UDP) Connection ID 0x405a75ad
Handshake
Src: 5.6.7.8:12345 Dst 45.83.174.13:443
(UDP) Connection ID 0xa901322a
Handshake
Src: 9.10.11.12:47385 Dst 45.83.174.13:443 (UDP)
Connection ID 0xb891148f
Src: 1.3.5.7:39485 Dst
45.83.174.13:443 (UDP)
Connection ID 0x74383bde
Src: 10.20.30.40:12345 Dst 45.83.174.13:443
(UDP) Connection ID 0x405a75ae
Src: 2.4.6.8:38473 Dst 45.83.174.13:443 (UDP)
Connection ID 0xa90187df
Time