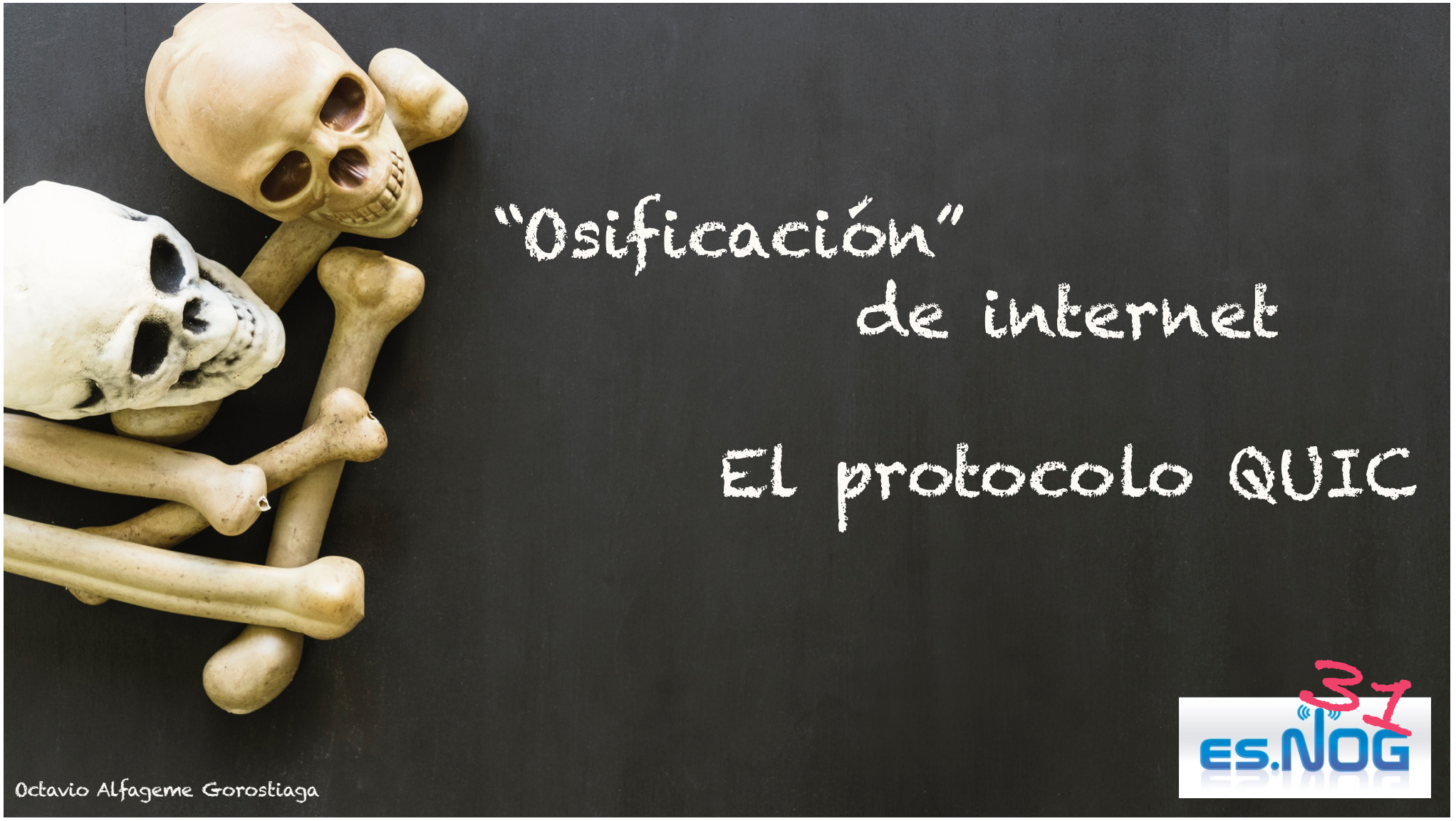
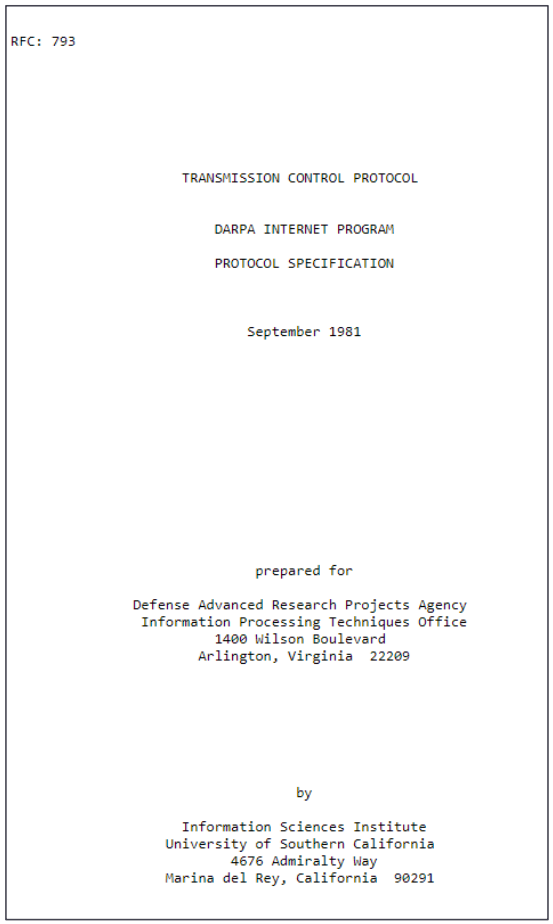

"Osificación"
de internet
El protocolo QUIC
Octavio Alfageme Gorostiaga
31
es.NOG

¿Qué pasó en 1981?



RFC: 793

TRANSMISSION CONTROL PROTOCOL

DARPA INTERNET PROGRAM

PROTOCOL SPECIFICATION

September 1981

prepared for

Defense Advanced Research Projects Agency
Information Processing Techniques Office
1400 Wilson Boulevard
Arlington, Virginia 22209

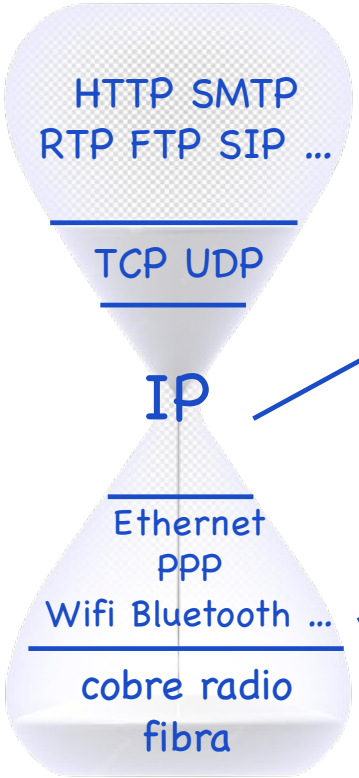
by

Information Sciences Institute
University of Southern California
4676 Admiralty Way
Marina del Rey, California 90291

Se publica la RFC793 Transmision Control Protocol

La ¿evolución? de TCP

El reloj de arena de internet – 43 años



Múltiples protocolos de capa de aplicación

¿Y las tecnologías de transporte?

IP como protocolo "core"

Evolución en las tecnologías de capa 2

Todos los medios físicos considerados

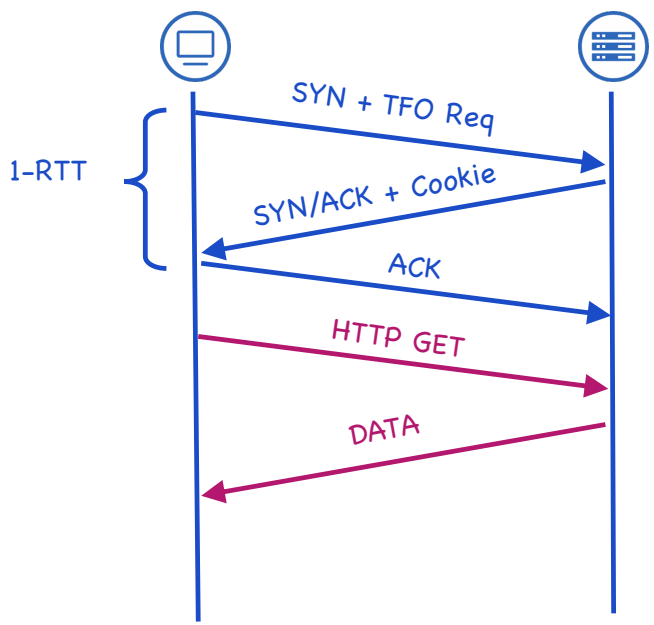
¿No ha evolucionado TCP?

- T/TCP – 1994
- TCP Session – 1997
- Congestion Manager – 1998
- SCTP – 2000
- SST (basado en UDP) – 2007
- Minion (basado en TCP y TLS) – 2011
- TCP Fast Open – 2009-2014
- MPTCP – 2013

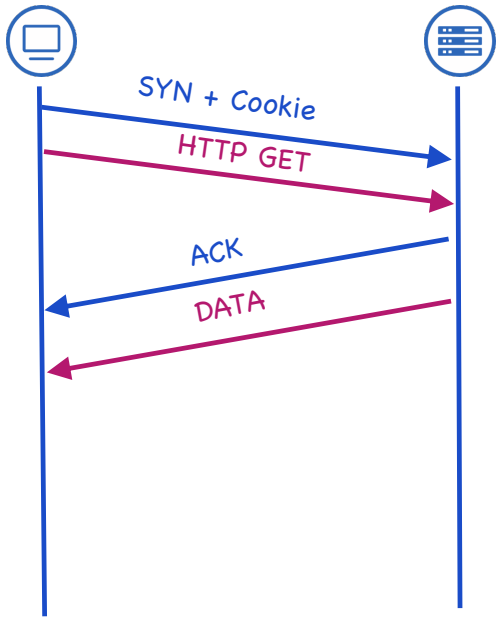
Reducir el número de "round-trips" en el handshaking de TCP

"Migración de conexión" / envío simultáneo de tráfico de la misma sesión TCP por dos interfaces diferentes

TCP Fast Open



Primera conexión - 1 RTT



Siguientes conexiones - 0 RTT

¿Por qué no evolucionó TCP?

"Osificación" de internet

"Middleboxes" (RFC3234 "Middleboxes: Taxonomy and Issues")
"defined as any intermediary box performing functions apart from normal, standard functions of an IP router on the data path between a source host and destination host"

- FW – IDS/IPS
- CGNAT
- Aceleradores/compresores
- Balanceadores de carga
- Routers y proxies

TCP en kernel del S.O. de todos los dispositivos conectados a internet

Es imposible evolucionar TCP (i.e. Fast Open)

El protocolo 146

¿Y si creamos un nuevo protocolo sobre IP? TCP 6, UDP 17, ...

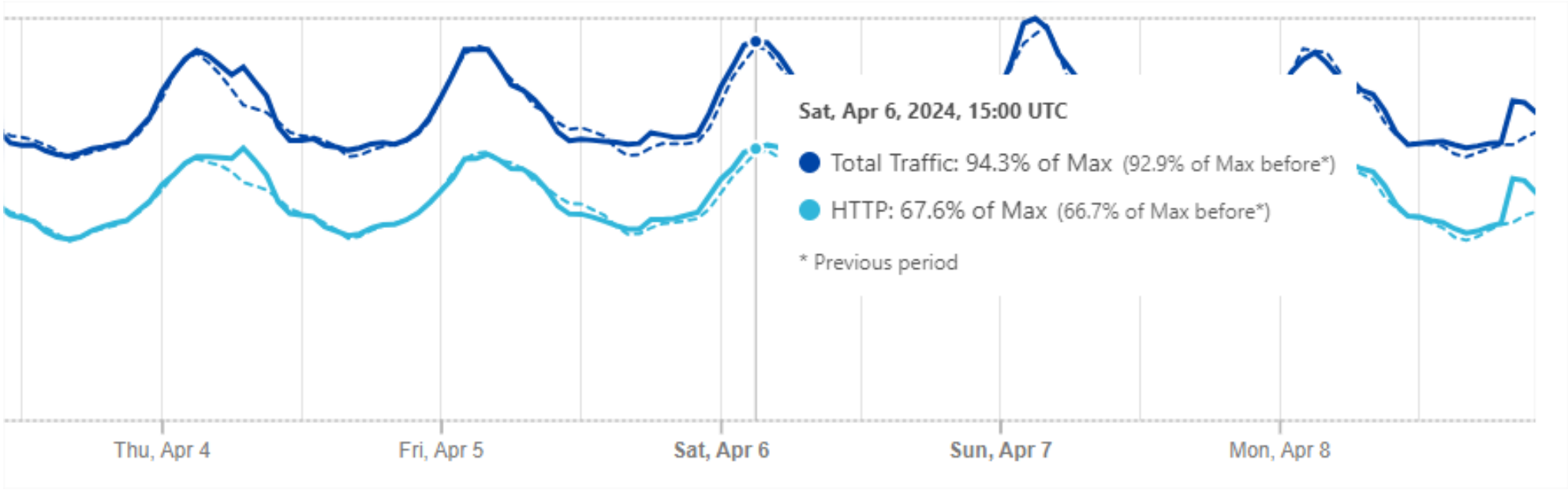
https://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml

138	manet	MANET Protocols		[RFC5498]
139	HIP	Host Identity Protocol	Y	[RFC7401]
140	Shim6	Shim6 Protocol	Y	[RFC5533]
141	WESP	Wrapped Encapsulating Security Payload		[RFC5840]
142	ROHC	Robust Header Compression		[RFC5858]
143	Ethernet	Ethernet		[RFC8986]
144	AGGFRAG	AGGFRAG encapsulation payload for ESP		[RFC9347]
145	NSH	Network Service Header	N	[RFC9491]
146-252		Unassigned		[Internet Assigned Numbers Authority]
253		Use for experimentation and testing	Y	[RFC3692]
254		Use for experimentation and testing	Y	[RFC3692]
255	Reserved			[Internet Assigned Numbers Authority]

¿Cuánto tardaríamos en desplegarlo en internet?

Internet en 2024

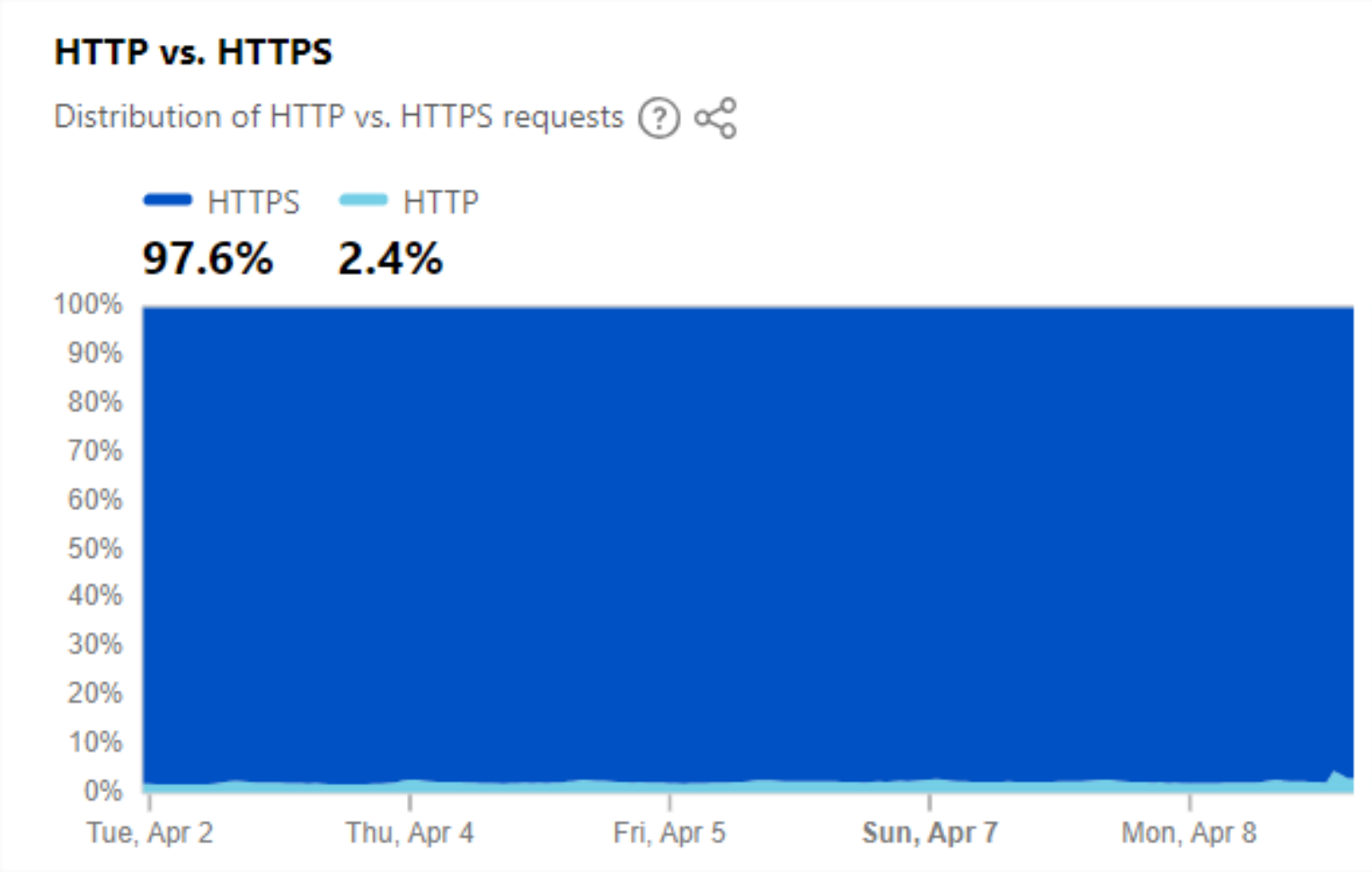
HTTP supone entre el 65% y 75%



Fuente: Cloudflare Radar

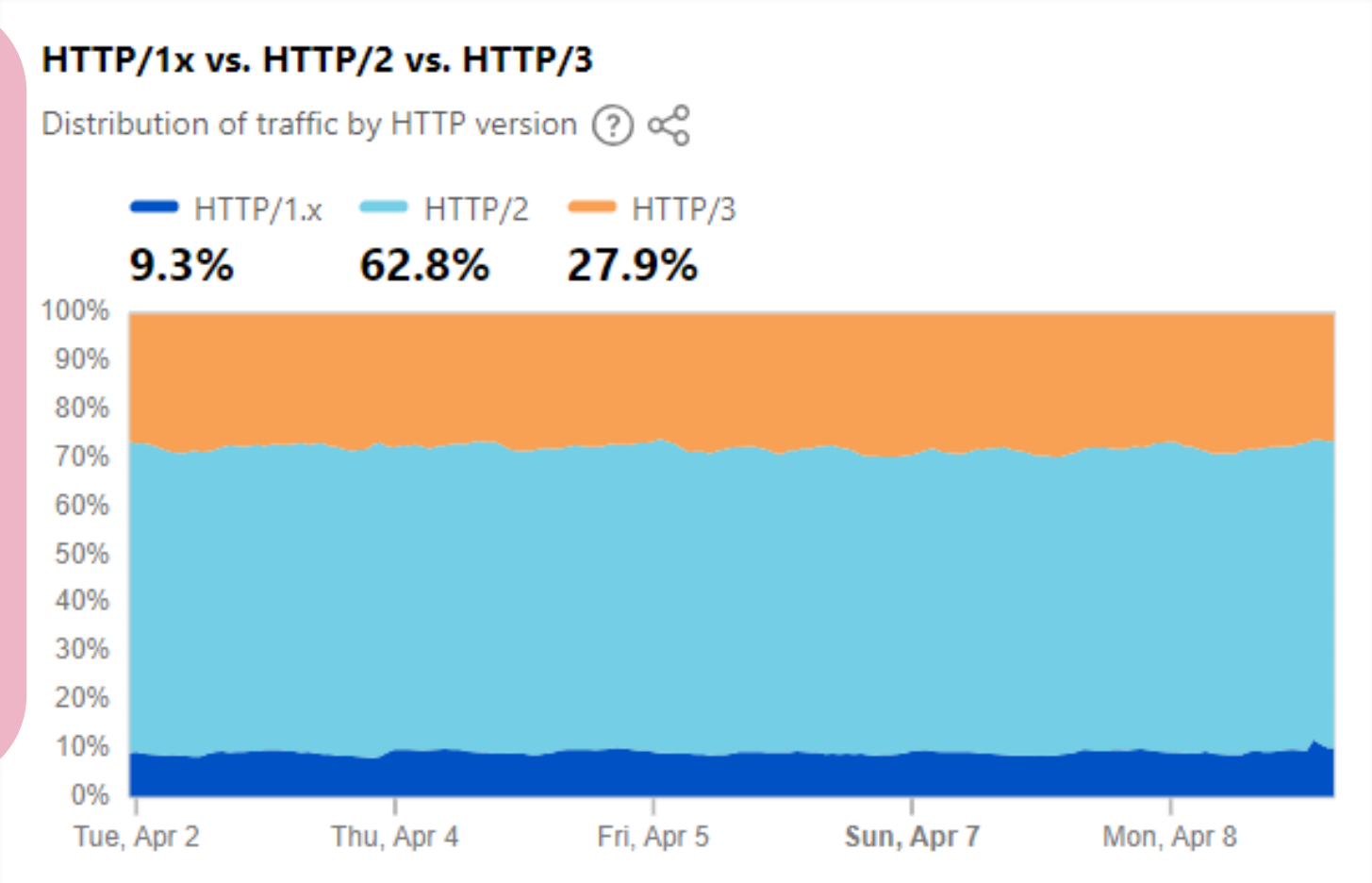
HTTPS es universal

Fuente: Cloudflare Radar



HTTP/3 (QUIC) se acerca al 30% de HTTP

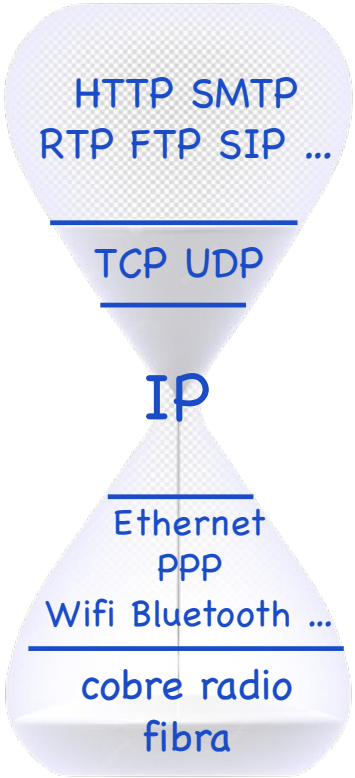
- HTTP/3 (sobre QUIC) más del 20% del tráfico total de internet
- Google Chrome, Mozilla Firefox, Microsoft Edge, Opera, Apple Safari, Android WebView, Brave, Vivaldi, …, intentan QUIC por defecto
- "Fallback" a TCP



Fuente: Cloudflare Radar

De HTTP/1.x a HTTP/3

Google – mejora de la experiencia

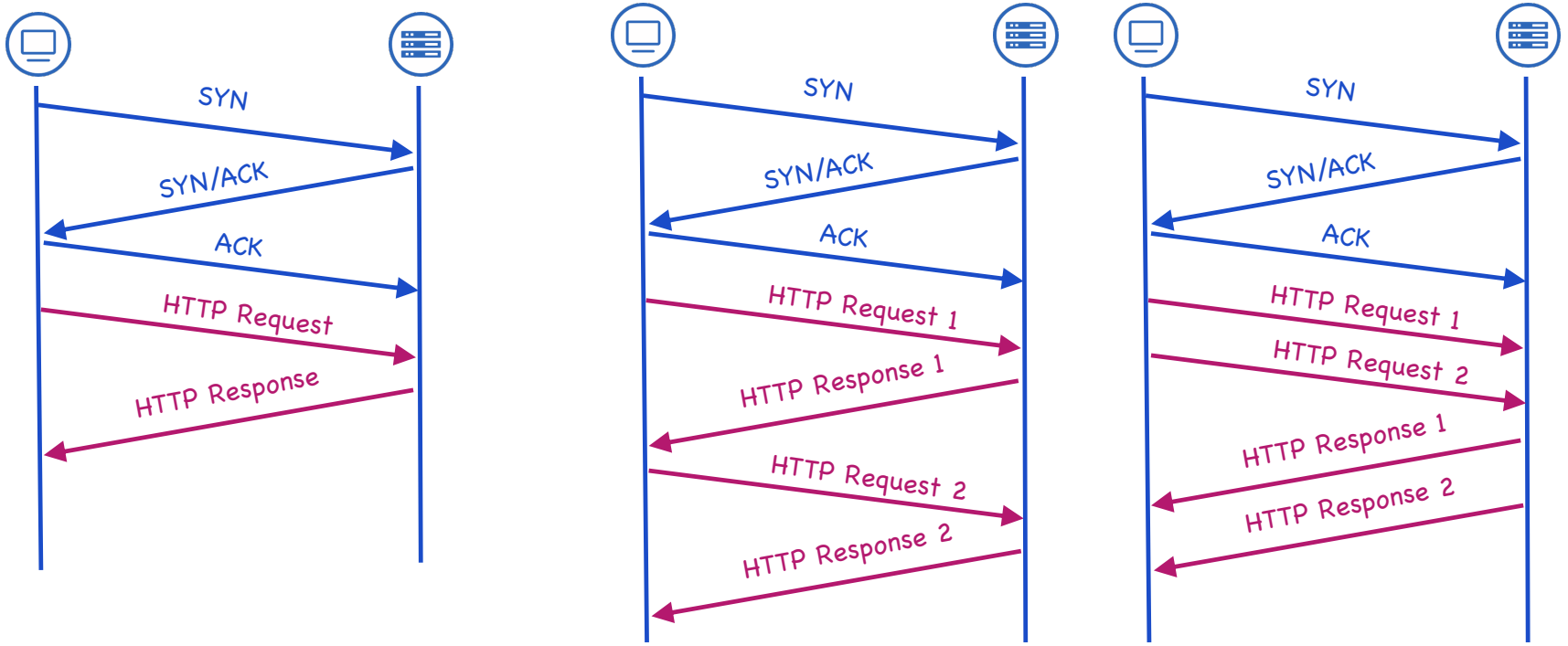


- Navegador propio (Google Chrome) y todo tipo de servidores y servicios como YouTube, google.com, ...
- Mejora de protocolos con iniciativas como SPDY

¿Cómo evolucionar el transporte sin ser víctima de la "osificación" de internet?

Google Global Cache (GGC) y fibra propia transoceánica y local

HTTP/1.X



HTTP/1.0
1996

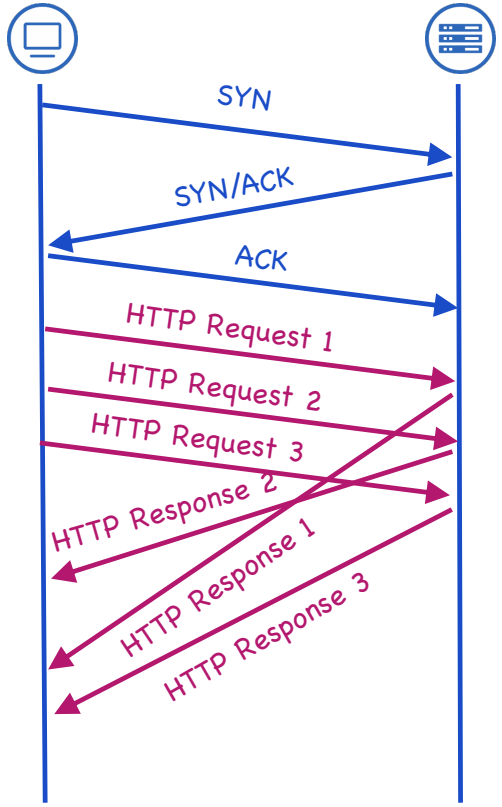
- sesión TCP por petición HTTP

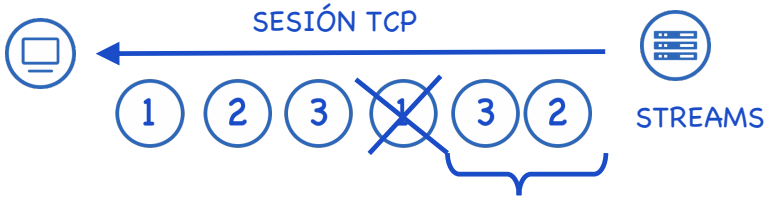
- persistencia de conexión
- pipelining
- "HoL blocking" a nivel HTTP (solicitudes atendidas en orden)

HTTP/1.1
1999

HTTP/2.0 - 2015

- SPDY como precursor
- Multiplexación de "streams"
- Sesión TCP única

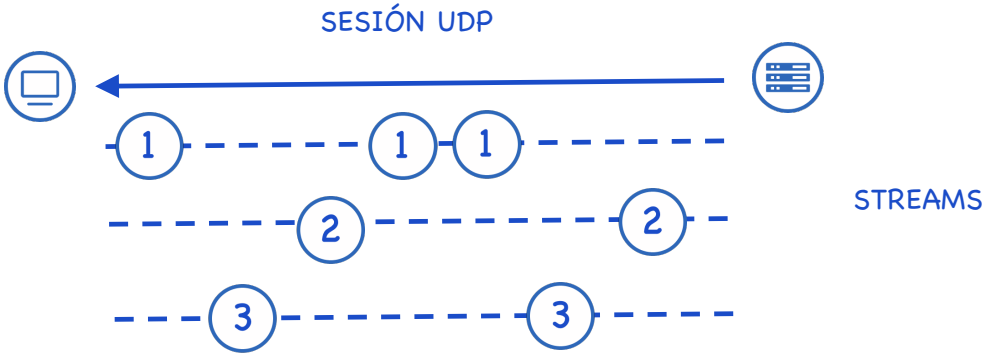




No se procesan hasta recibir el paquete perdido del stream 1

- "HoL blocking" a nivel TCP

HTTP/3.0 (QUIC) - RFC9114 - 2022



- QUIC posibilita múltiples "streams" paralelos - retransmisión a nivel de "stream" (no "HoL blocking")
- Incorpora las ventajas de TLS 1.3 (handshaking, cipher suites, 0-RTT, ...)

QUIC - ¿el fin de la "osificación"?

Origen de QUIC

- Google desarrolló gQUIC → ofrecer a HTTP una comunicación sin el "HoL blocking" de HTTP/2
- Desplegado en Chrome y sus servicios en 2014 → hasta 7% de internet
- En 2016 compartido con el IETF → QUIC Working Group

2021

- RFC 8999 - Version-Independent Properties of QUIC
- RFC 9000 - QUIC: A UDP-Based Multiplexed and Secure Transport
- RFC 9001 - Using TLS to Secure QUIC
- RFC 9002 - QUIC Loss Detection and Congestion Control

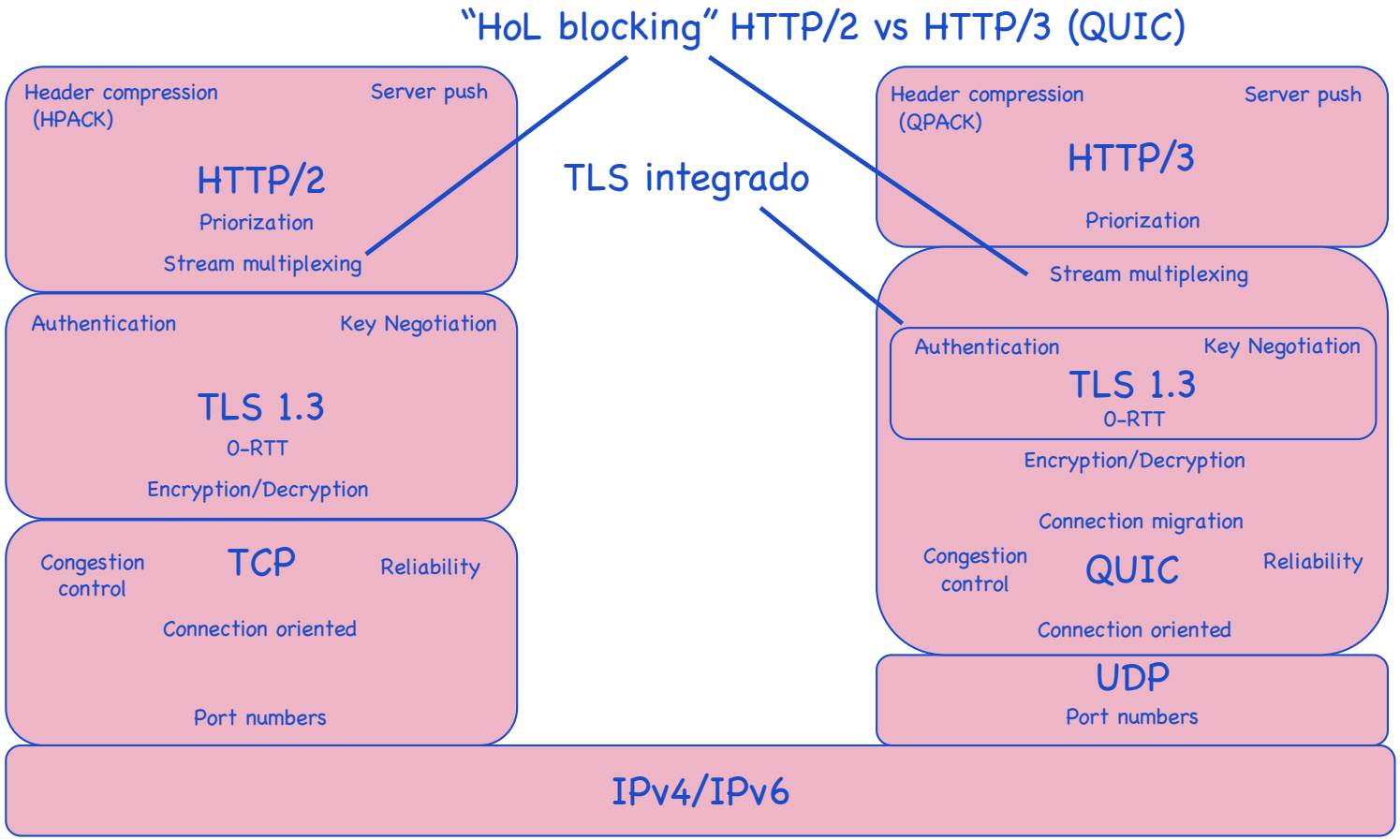
Objetivos de QUIC

1. Evolucionable – no sujeto a "osificación"
2. Baja latencia en establecimiento de conexión
3. Fin del "HoL blocking" de HTTP/2
4. Mejora de mecanismos de control de congestión y retransmisión de tráfico
5. "Migración de conexión" nativa

QUIC nació vinculado a HTTP/3, pero es un protocolo de transporte genérico

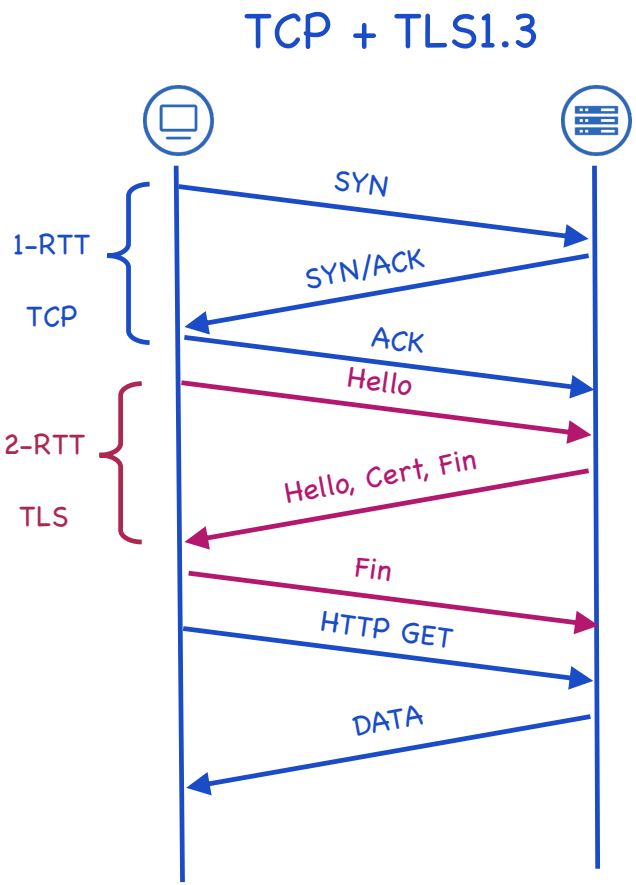
QUIC sobre UDP

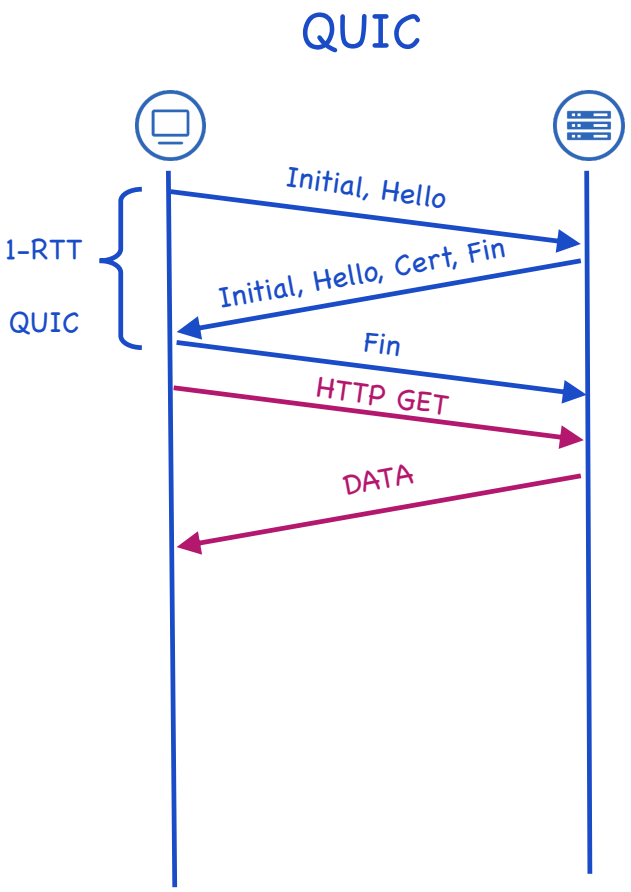
- UDP → "antiosificación"



1-RTT

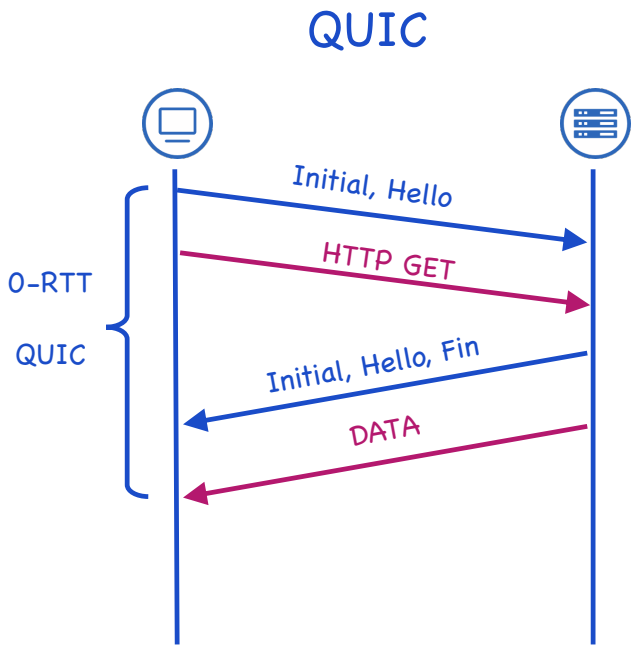
- Integración de TLS permite acortar el "handshaking"
- Apreciable en descargas cortas





0-RTT

- "Session resumption" – válido para sesiones previamente establecidas
- Funcionalidad de TLS 1.3

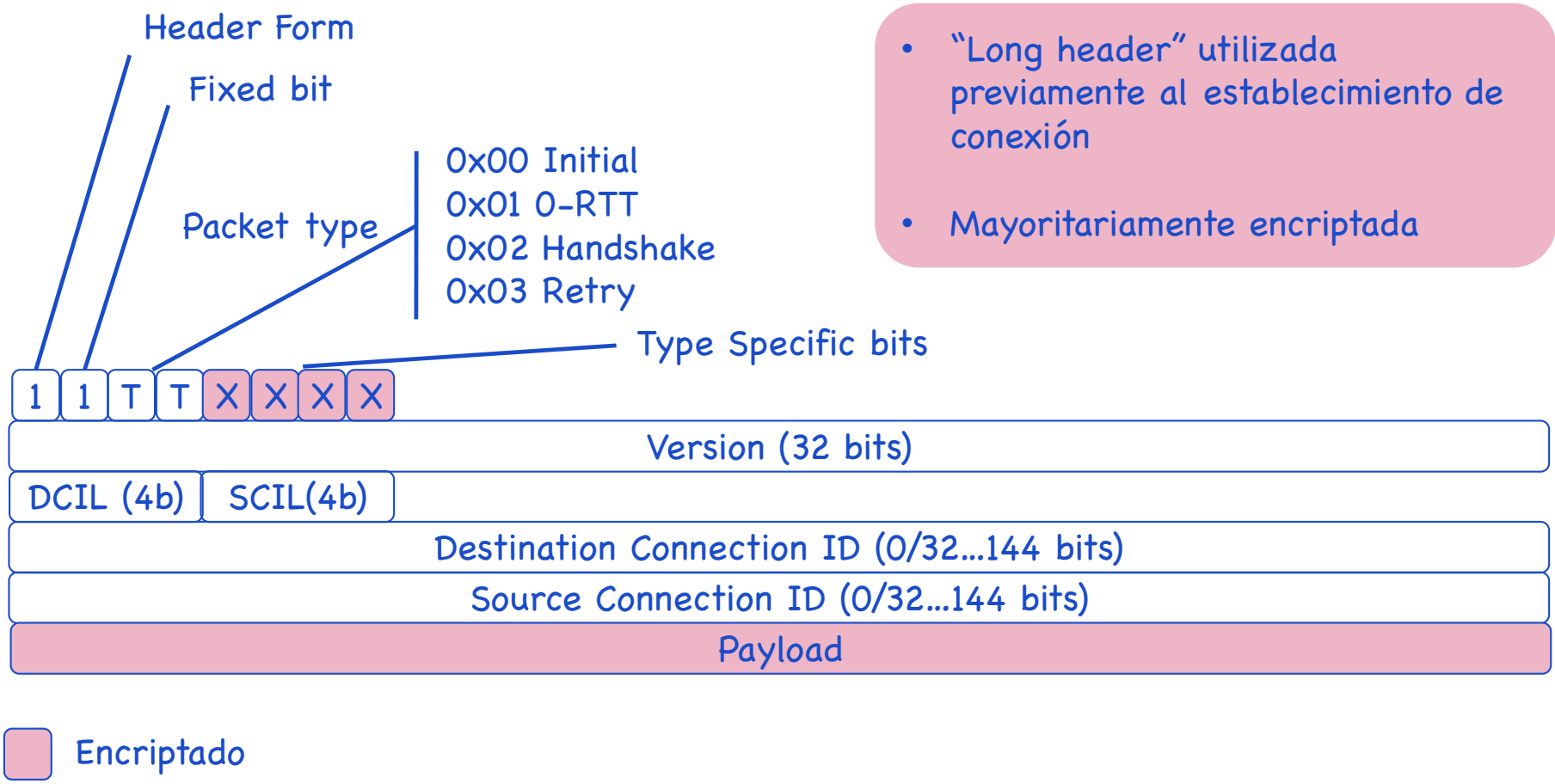


No.	Time	Source	Destination	Protocol	Length	Info
18	0.983820	192.168.2.106	142.250.201.74	QUIC	1292	Initial, DCID=b1e062224e485d7a, PKN: 1, PING, CRYPTO, PADDING, PING, PING, CRYPTO, PING, PING, P
19	0.983950	192.168.2.106	142.250.201.74	QUIC	120	0-RTT, DCID=b1e062224e485d7a
25	1.005647	192.168.2.106	142.250.201.67	QUIC	1292	Initial, DCID=8521f4ae45f0229f, PKN: 1, CRYPTO, CRYPTO, CRYPTO, PING, PADDING, CRYPTO, CRYPTO, C
29	1.015539	192.168.2.106	142.250.184.174	QUIC	1292	Initial, DCID=71bb4658af1c31fd, PKN: 1, CRYPTO, CRYPTO, CRYPTO, PING, PING, PING, CRYPTO, CRYPTO
30	1.015670	192.168.2.106	142.250.184.174	QUIC	123	0-RTT, DCID=71bb4658af1c31fd
34	1.031478	192.168.2.106	34.160.226.139	QUIC	1292	Initial, DCID=c2f22765dd0460eb, PKN: 1, PADDING, CRYPTO, CRYPTO, CRYPTO, CRYPTO, PADDING, CRYPTO
41	1.033123	142.250.201.74	192.168.2.106	QUIC	1292	Handshake, SCID=f1e062224e485d7a
42	1.033124	142.250.201.74	192.168.2.106	QUIC	852	Protected Payload (KP0)
43	1.033124	142.250.201.74	192.168.2.106	QUIC	238	Protected Payload (KP0)

"Connection-ID"

- QUIC no se apoya en la tupla "IP origen, IP destino, puerto origen, puerto destino"
- "Connection-ID" unidireccional generado aleatoriamente por cliente y servidor

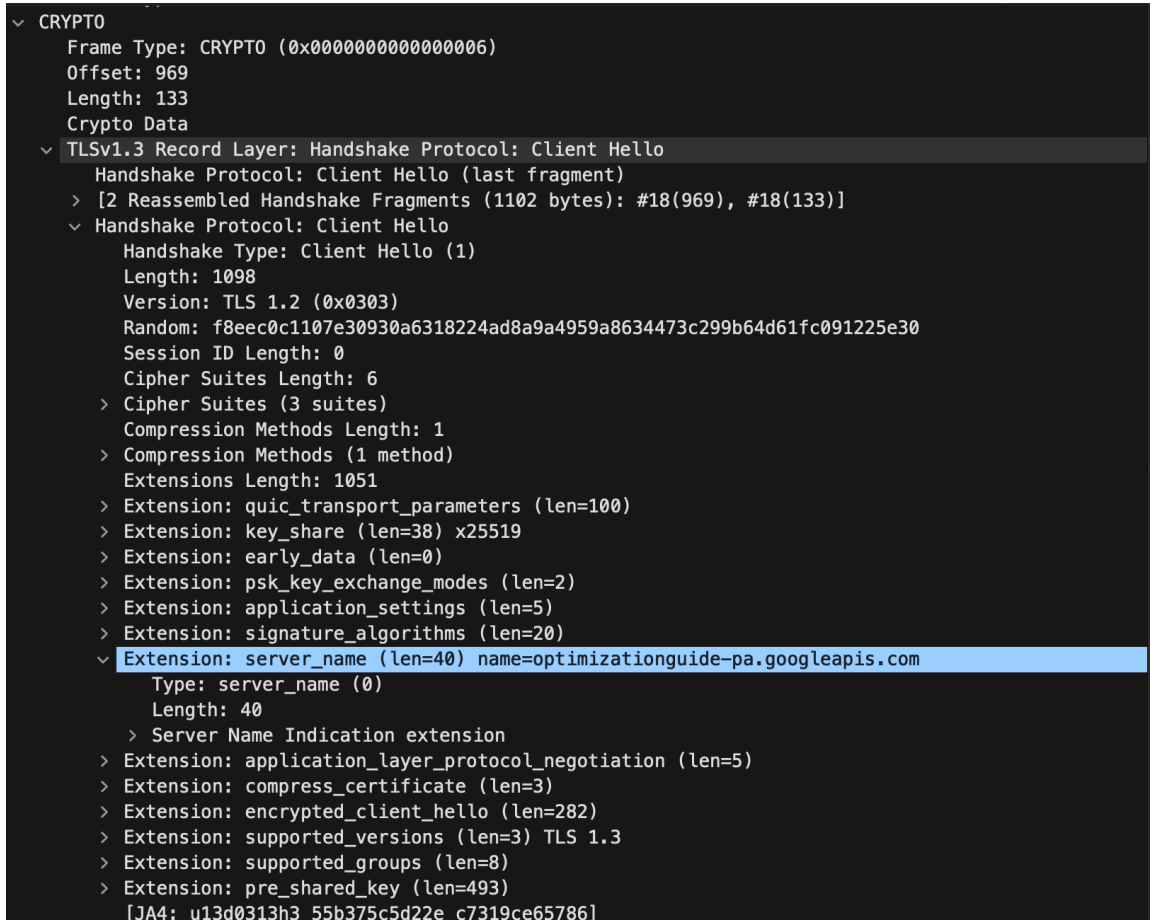
Cabecera QUIC – "Long header"



- "Long header" utilizada previamente al establecimiento de conexión
- Mayoritariamente encriptada

Encriptado

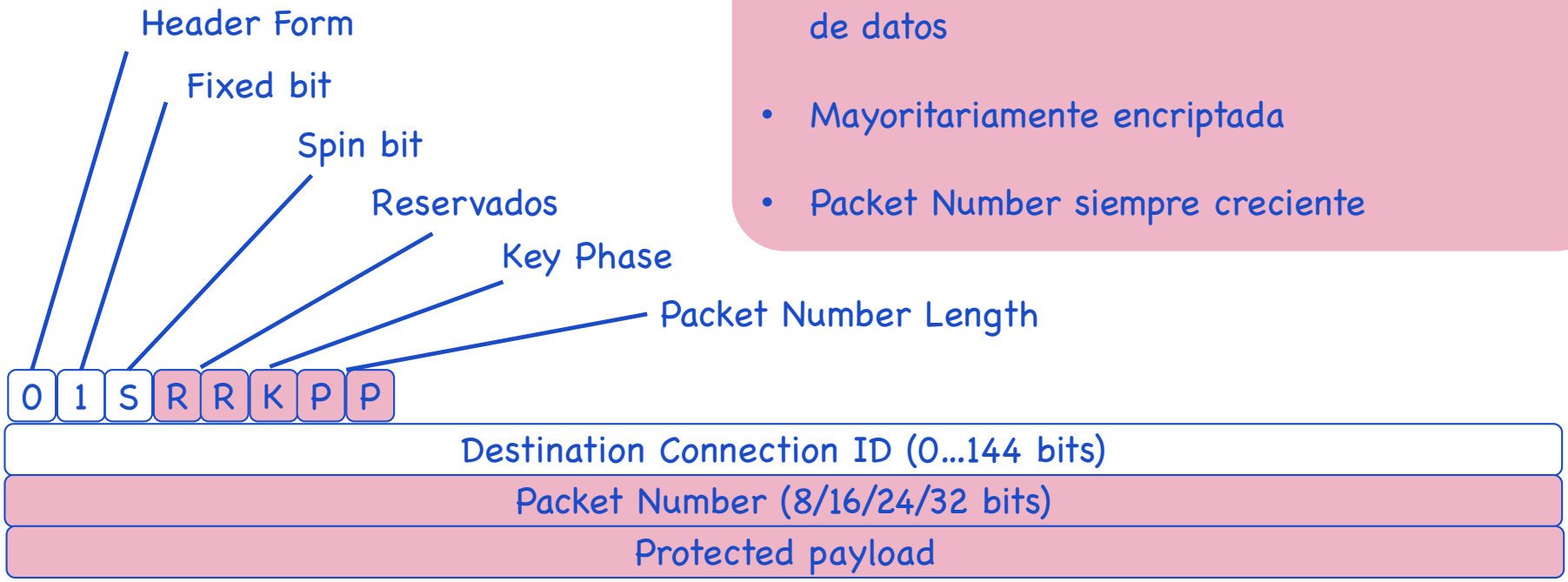
Cabecera QUIC – "Long header"



- No todo el payload va encriptado
- El "client hello" de TLS va abierto – contiene todas las extensiones soportadas por el cliente (en trama CRYPTO)

Cabecera QUIC – "Short header"

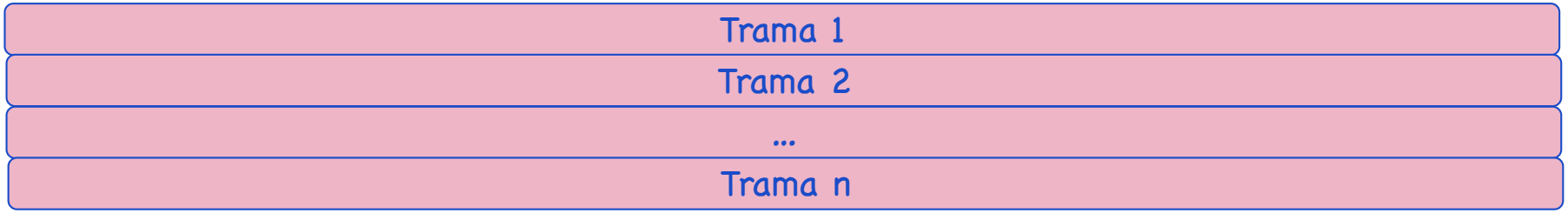
- "Short header" utilizada en la transferencia de datos
- Mayoritariamente encriptada
- Packet Number siempre creciente



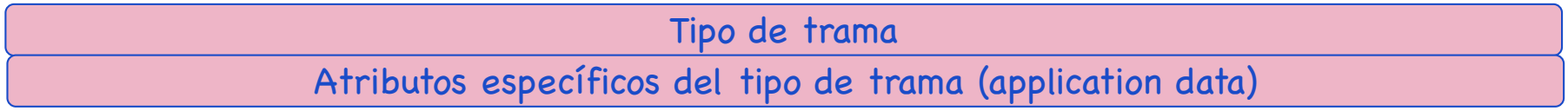
Encriptado

Tramas QUIC

- El payload de los paquetes QUIC se dividen en tramas



- Cada tipo de trama tiene su estructura



Encriptado

Tipos de tramas QUIC

- Tramas tanto de control como de datos
- Unidireccional vs bidireccional
- Client vs server initiated
- Tres tramas principales

Type Value	Frame Type Name
0x00	PADDING
0x01	PING
0x02 - 0x03	ACK
0x04	RESET_STREAM
0x05	STOP_SENDING
0x06	CRYPTO
0x07	NEW_TOKEN
0x08 - 0x0f	STREAM
0x10	MAX_DATA
0x11	MAX_STREAM_DATA
0x12 - 0x13	MAX_STREAMS
0x14	DATA_BLOCKED
0x15	STREAM_DATA_BLOCKED
0x16 - 0x17	STREAMS_BLOCKED
0x18	NEW_CONNECTION_ID
0x19	RETIRE_CONNECTION_ID
0x1a	PATH_CHALLENGE
0x1b	PATH_RESPONSE
0x1c - 0x1d	CONNECTION_CLOSE

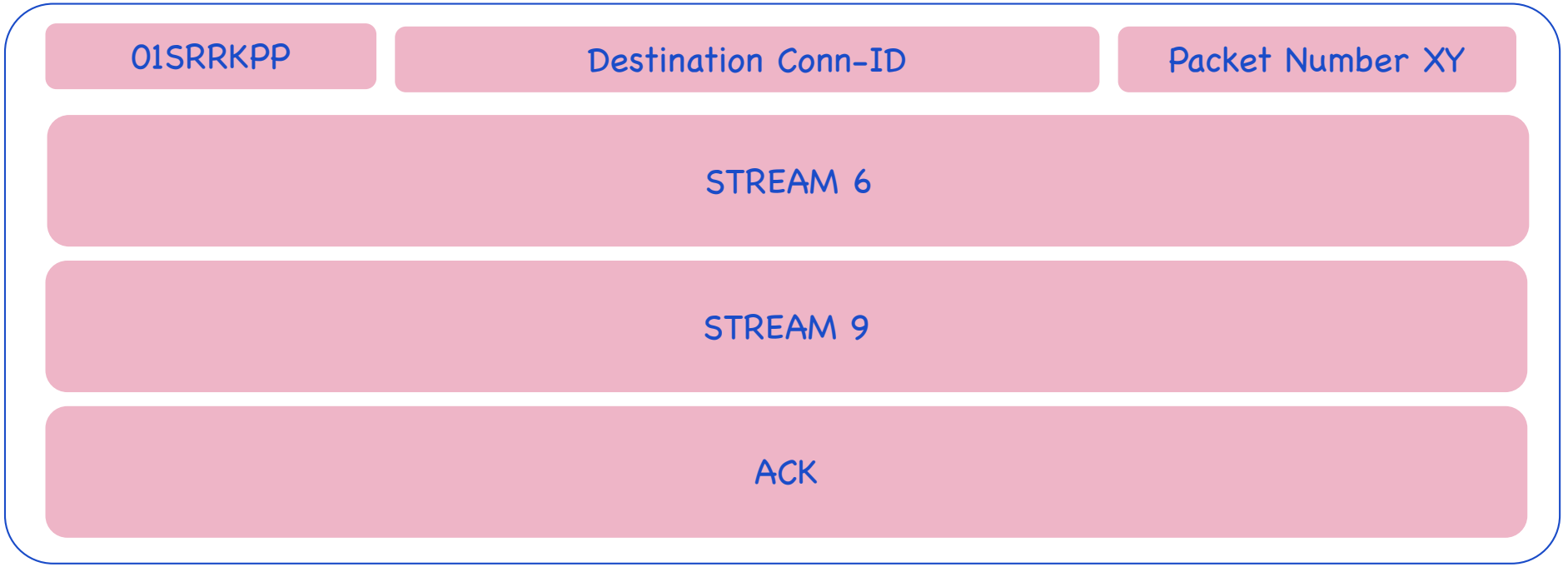
- Las tramas "stream" llevan los datos y permiten mantener flujos paralelos y así terminar con el "HoL blocking" de HTTP/2
- Las tramas "crypto" se utilizan en el establecimiento de la conexión segura
- Las tramas "ACK" permiten la confirmación de recepción de paquetes QUIC

Paquete QUIC

- Integra múltiples tramas en el mismo paquete
- Múltiples flujos

```
QUIC IETF
  QUIC Connection information
      [Connection Number: 1]
    [Packet Length: 80]
  > QUIC Short Header PKN=12672304
  > STREAM id=3 fin=0 off=0 len=24 dir=Unidirectional origin=Server-initiated
  > STREAM id=7 fin=0 off=0 len=1 dir=Unidirectional origin=Server-initiated
  > STREAM id=11 fin=0 off=0 len=1 dir=Unidirectional origin=Server-initiated
  > PADDING Length: 24
```

```
QUIC IETF
  > QUIC Connection information
    [Packet Length: 35]
  > QUIC Short Header DCID=efbf8f48a7991a0a PKN=7
  > ACK
  > STREAM id=6 fin=0 off=0 len=2 dir=Unidirectional origin=Client-initiated
```



Gestión de tráfico

MTU

- RFC9000
  - No existe fragmentación
  - Asume que la red admite 1280 bytes IP
  - PMTUD para "datagramas" QUIC de más de 1200 bytes (no implementado actualmente)

Control de flujo

- Evolución sobre el de TCP
- Dos niveles:
  - Trama "STREAM"
  - Agregado de conexión – control del buffer de todos los "STREAMs" en el receptor

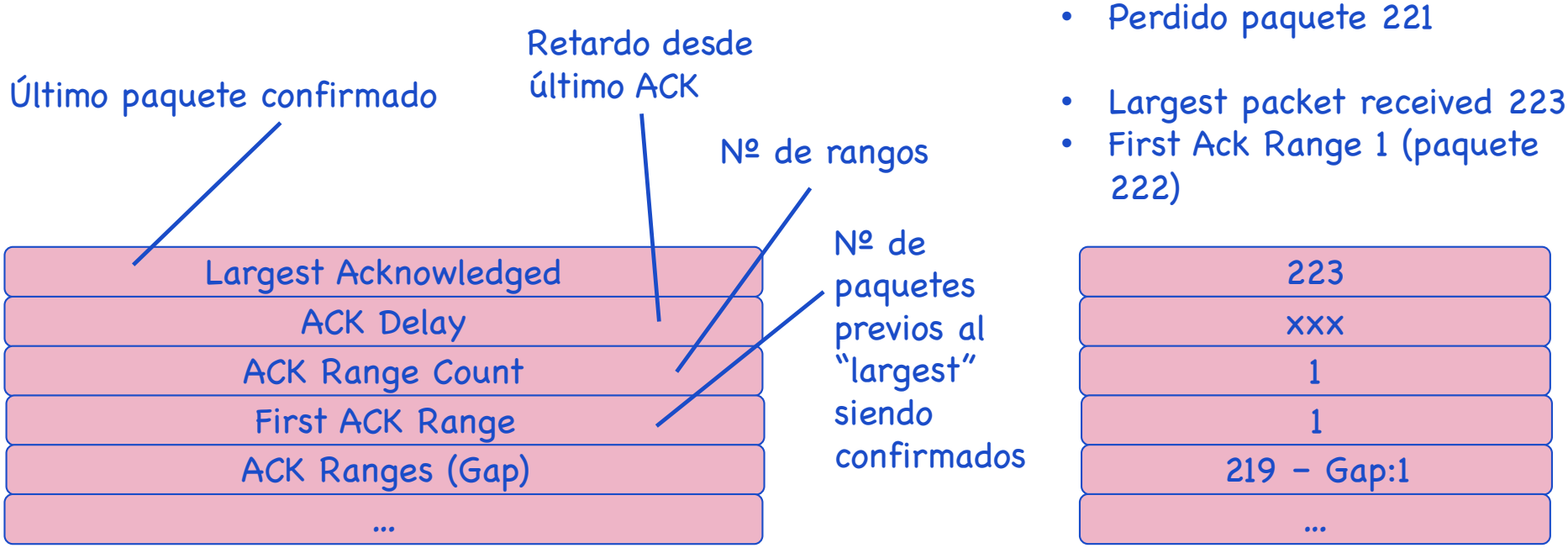
Gestión de tráfico (II)

Control de congestión y recuperación

- Estimación del RTT
  - Cada extremo lo usa en el detección de pérdida de trama
- Detección de pérdida
  - ACKs confirman la recepción de paquetes QUIC (no por trama)
  - PTO (Probe Timeout) para detección de pérdida de paquete
- Control de la congestión
  - RFC9002 selecciona NewReno, permitiendo extensiones
  - Google Chrome soporta hoy Cubic y BBR

Gestión de tráfico – tramas ACK

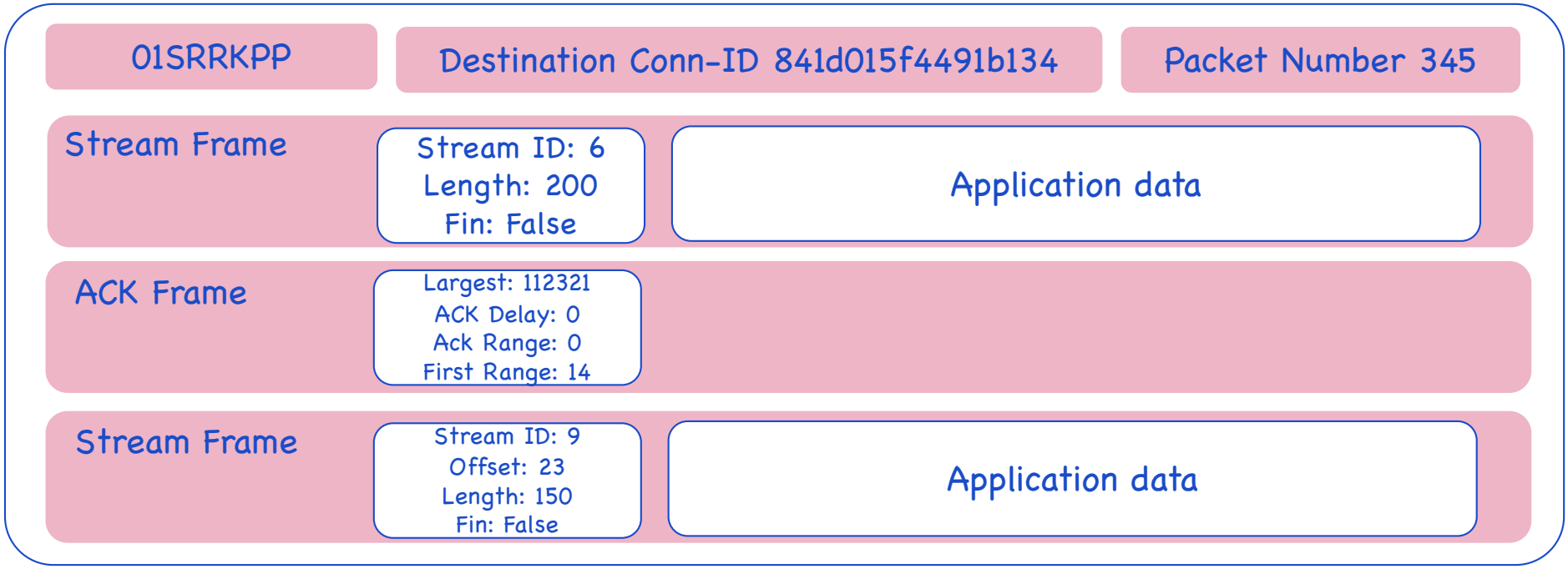
- Es posible confirmar rangos de paquetes QUIC e indicar pérdidas puntuales de paquetes (ACK Range y Gap)



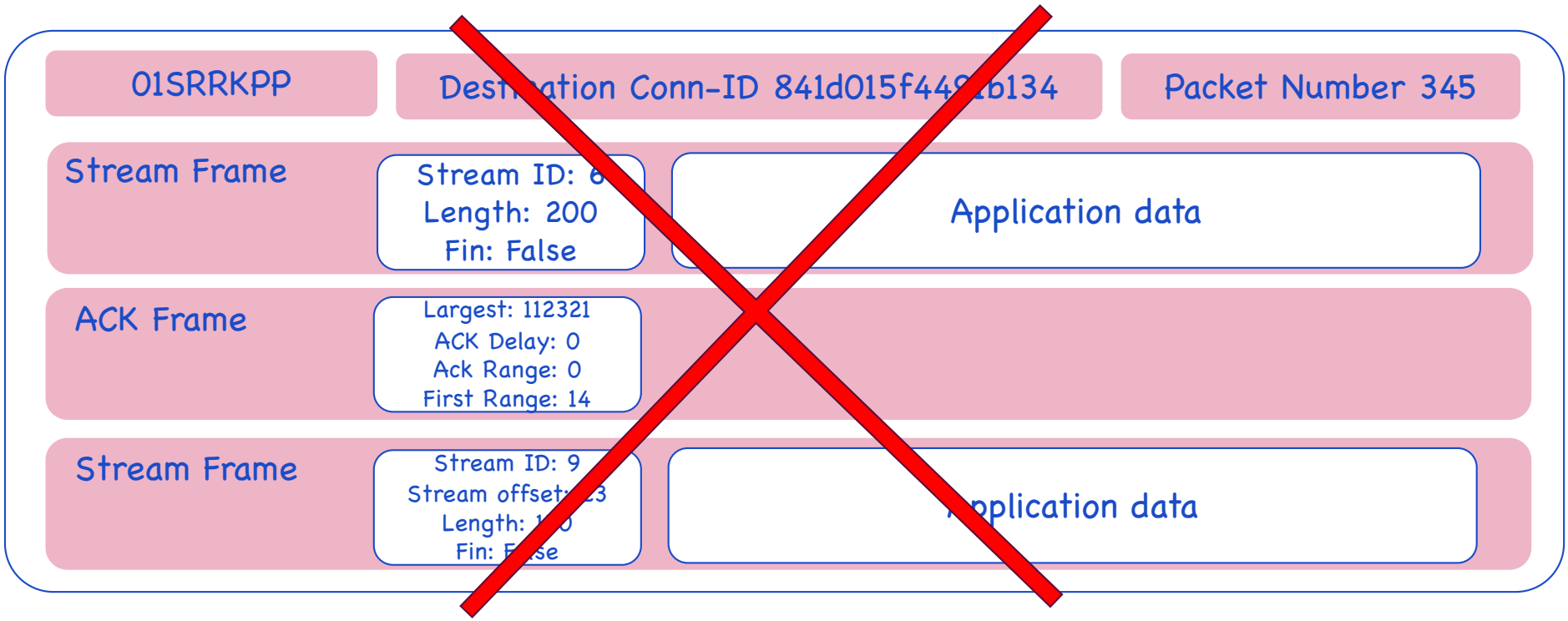
- Perdido paquete 221

- Largest packet received 223
- First Ack Range 1 (paquete 222)

Ejemplo – pérdida de paquete

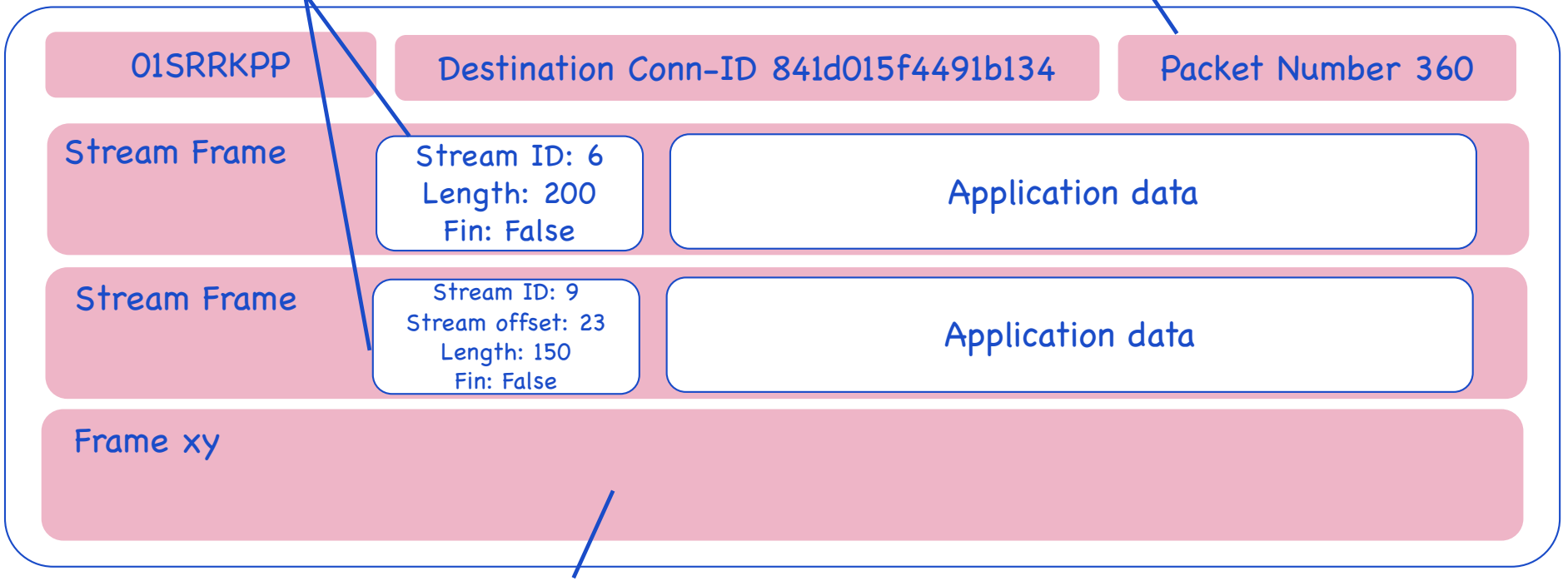


Ejemplo – pérdida de paquete (II)



Ejemplo – pérdida de paquete (III)

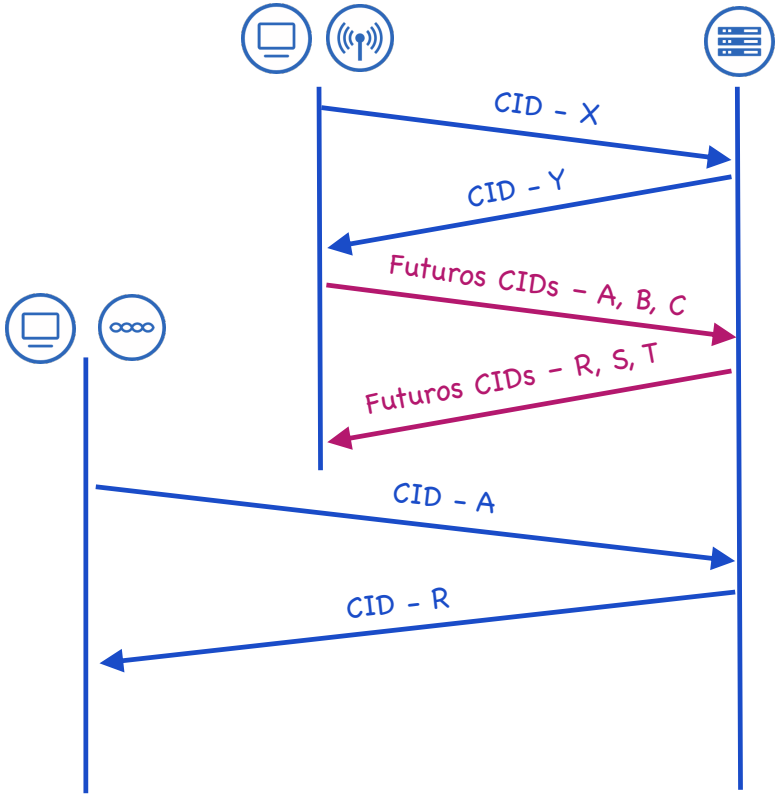
- Retransmisión de las tramas perdidas (el ACK se habrá "cubierto" en tramas posteriores
- "Packet number" incremental



- El paquete puede incorporar otras tramas

Migración de conexión

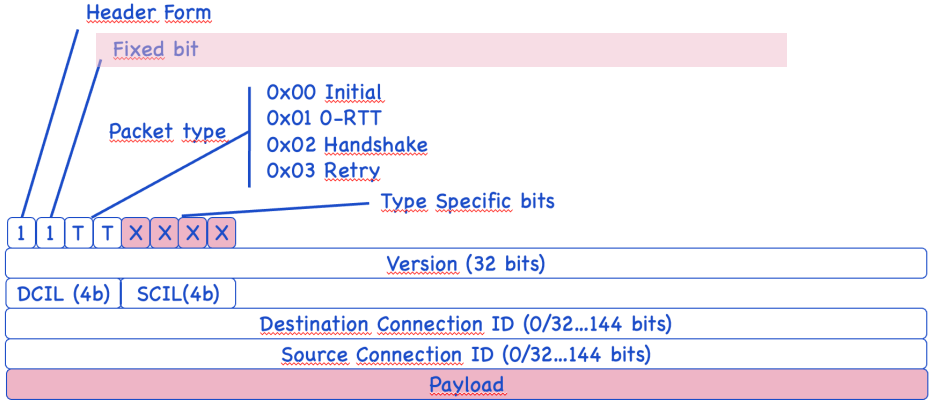
- Uno de los objetivos de QUIC
- Futuros "connection-IDs" intercambiados
- El cliente que migra de conexión modifica su "connection-ID"
- El servidor responde cambiando el suyo
- Futuros "connection-IDs" intercambiados van encriptados – no es posible seguir la sesión



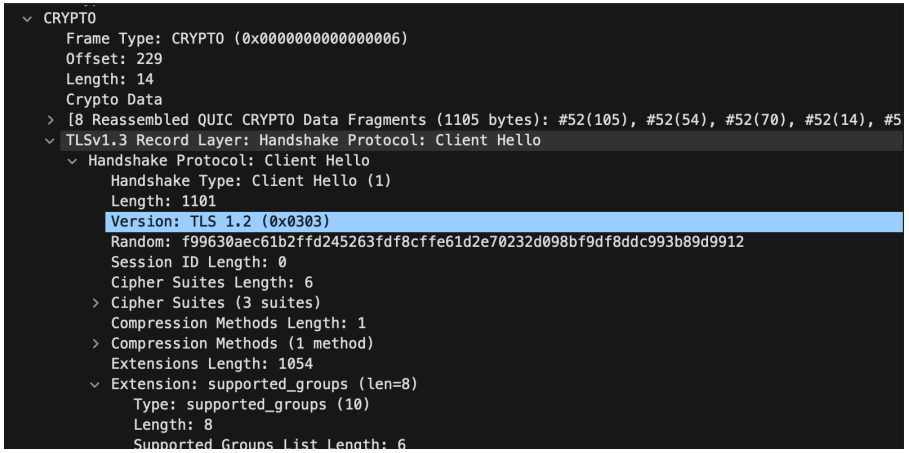
QUIC y los "middleboxes"

¿Es QUIC inmune a la "osificación"?

- Google sufrió la "osificación" de internet con gQUIC
- QUIC busca evitarlo al máximo
- RFC9287 "Greasing the QUIC bit" → mecanismo para modificar "aleatoriamente" el valor del "fixed bit" de la cabecera QUIC



- Ni siquiera TLS1.3 es inmune a ella



¿Supone QUIC el fin de la "osificación"?

- Va a seguir habiendo firewalls cortando el UDP 443, aceleradores que lo corten para "aprovechar" TCP, ...

Sólo la necesidad de aprovechar las ventajas de QUIC favorecerá este proceso

QUIC y los balanceadores

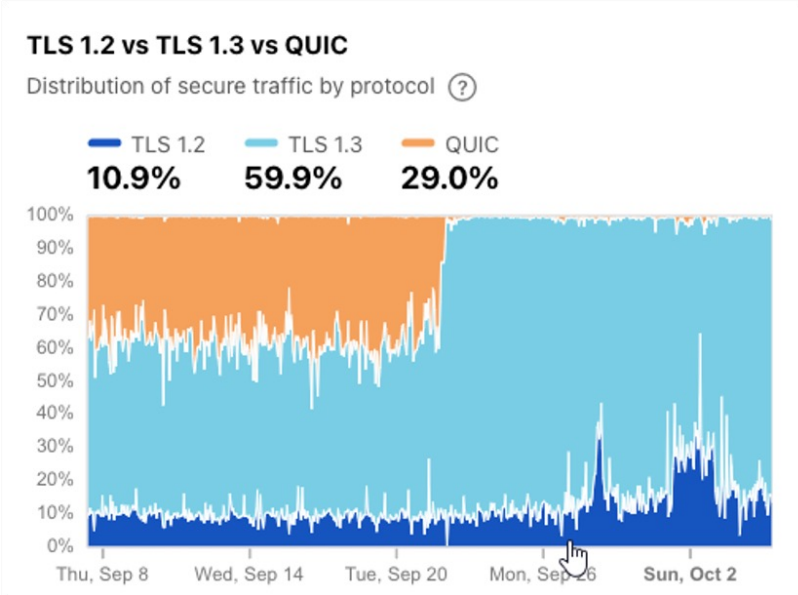
- Los balanceadores reparten carga entre cliente y servidor apoyados en IPs/puertos TCP/UDP origen y destino
- ¿Cómo hacerlo con conexiones QUIC en que existen "migraciones de conexión"?

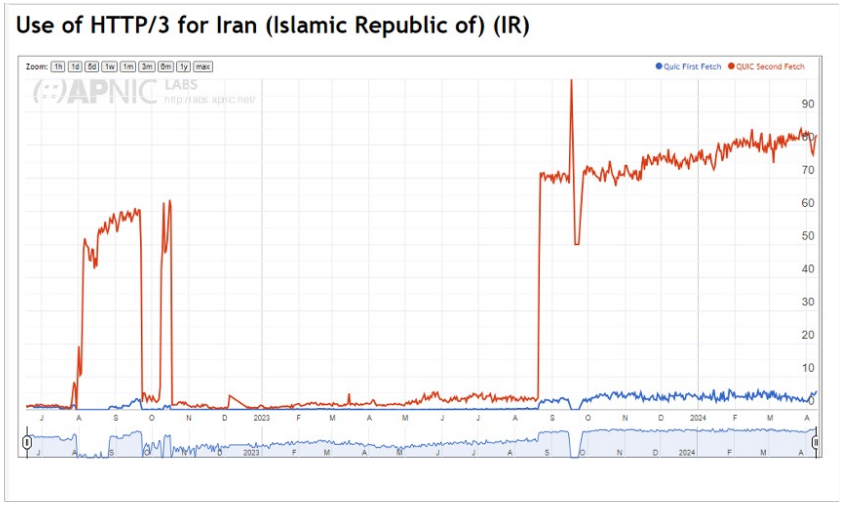
- El balanceador termina el HTTP/3 y usa HTTP/2 o HTTP/3 hacia el servidor

- draft-ietf-quic-load-balancers
- El servidor genera un "connection-id" enrutable por el balanceador

QUIC vs los firewalls nacionales

- Algunos países están probando a cortar QUIC puntualmente o por periodos más largos → Irán





Fuente: Cloudflare Radar y APNIC

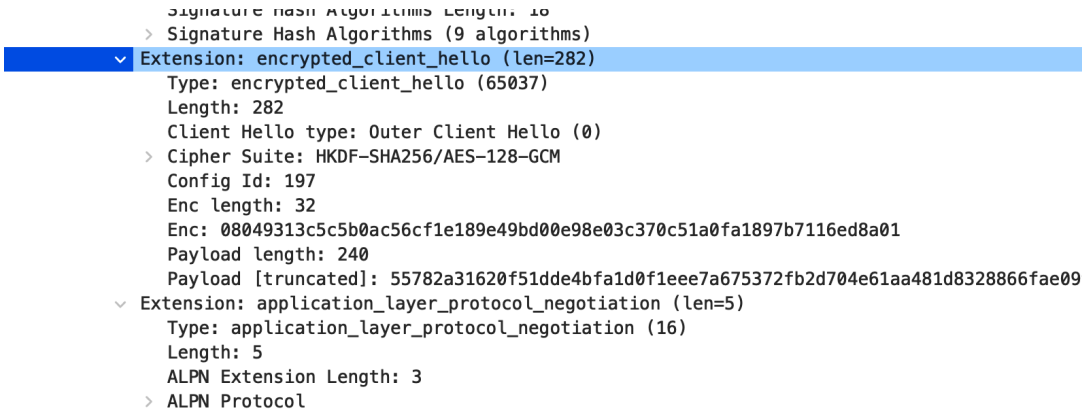
¿Cómo "cerrar" QUIC? - DoH/DoT + ECH

- DNS over HTTPS / DNS over TLS
- RFC8484 DNS Queries over HTTPS /RFC 7858 Specification for DNS over Transport Layer Security (TLS)

La respuesta DNS incluye clave pública

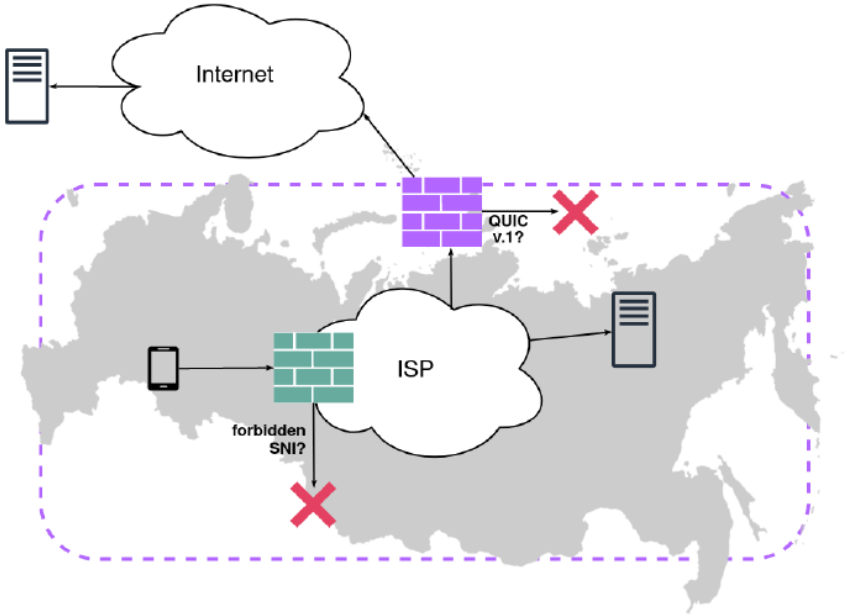
- ECH - Encrypted Client Hello
- draft-ietf-tls-esni TLS Encrypted Client Hello

Encriptación del "Client Hello"

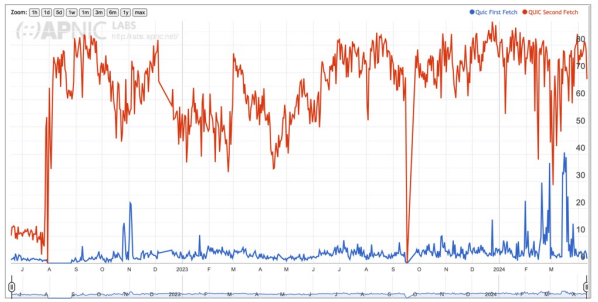


QUIC vs los firewalls nacionales (II)

- ECH es un esfuerzo en desarrollo
- DoH/DoT se puede bloquear/entorpecer



Use of HTTP/3 for Russian Federation (RU)



- Rusia corta el QUIC previo a v1 (draft29)
- Analiza el SNI (Server Network Identificator) en el "Client Hello" de QUIC (sin ECH)
- Filtra si es un destino "no autorizado"

Fuente: OONI – Open Observatory of Network Interference

https://ooni.org/post/2022-quick-look-quic-censorship/
https://ooni.org/post/2022-doh-dot-paper-dnsprivacy21/

Monitorización de QUIC

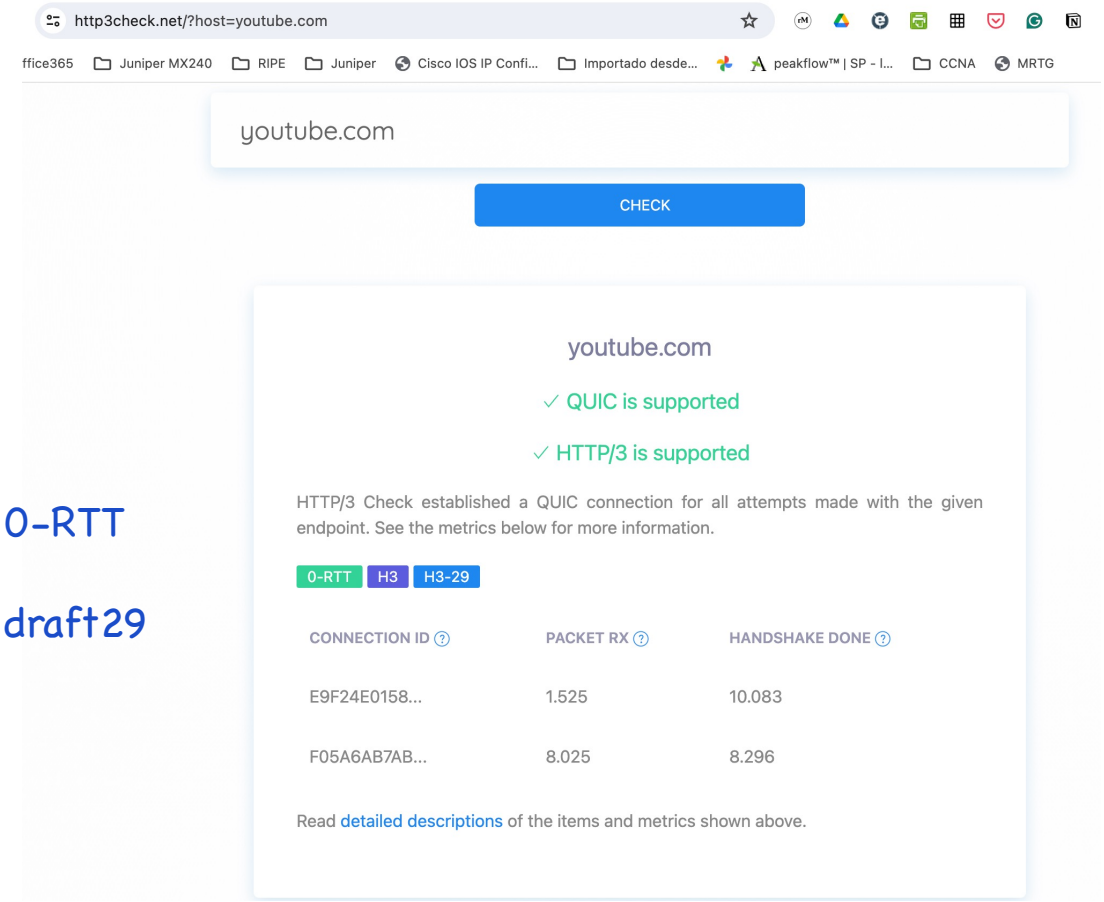
SPIN bit - RFC9000

- La encriptación QUIC dificulta cualquier medida pasiva
- Medida pasiva de latencia del RTT a partir del "connection-id" y del "spin-bit"


```
> Internet Protocol Version 4, Src: 192.168.2.106, Dst: 34.160.226.139
> User Datagram Protocol, Src Port: 54192, Dst Port: 443
v QUIC IETF
  v QUIC Connection information
      [Connection Number: 2]
    [Packet Length: 33]
  v QUIC Short Header DCID=edc40c98d015faca
      0... .... = Header Form: Short Header (0)
      .1.. .... = Fixed Bit: True
      ..0. .... = Spin Bit: False
      Destination Connection ID: edc40c98d015faca
    Remaining Payload: be2fe035758bd9ba22657beb8197d9fb7d1fcfb3799cfb98
```

Verificación de soporte de HTTP3

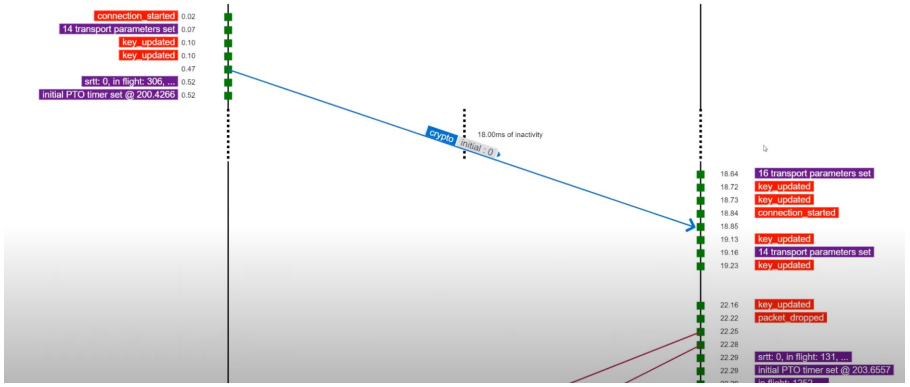
- HTTP3check

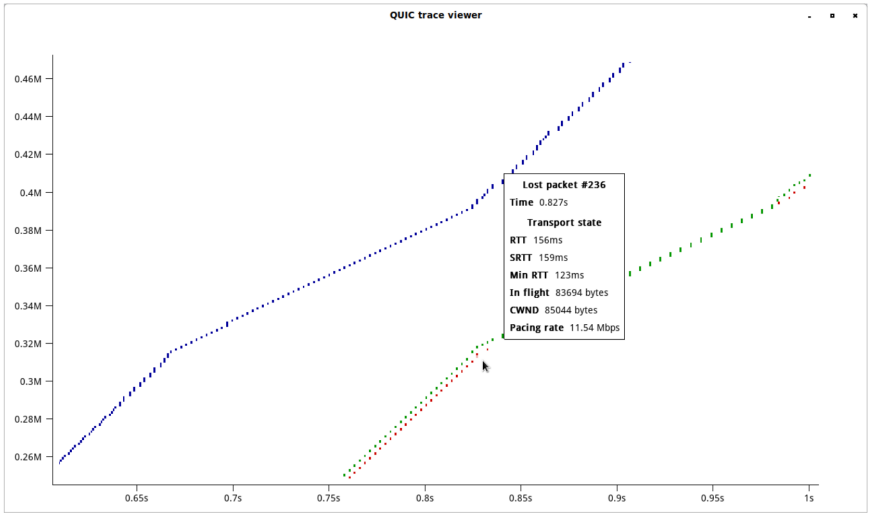


- Soporte de 0-RTT
- QUICv1
- Soporte de draft29

Análisis en host final

- Qvis
- QUIC-trace





What's next?

¿Y qué sucede con UDP?

- Es posible establece "streams" QUIC sin confirmación de entrega

RFC9221 "An Unreliable Datagram Extension to QUIC"

BGP over QUIC

- ¿Podrá suponer el final de BGP4 con "hash" MD5 o TCP-AO, así como "best-practices" de seguridad como GTSM (Generalized TTL Security Mechanism)?

draft-retana-idr-bgp-quic BGP over QUIC

Multicast extension for QUIC

- Apoyado en la RFC9221 es una alternativa al envío de tráfico multicast sobre red IP típico de servicios como el IPTV

draft-jholland-quic-multicast Multicast Extension for QUIC

...

- Using NETCONF over QUIC Connection - draft-dai-netconf-quic-netconf-over-quic
- DNS over Dedicated QUIC Connections RFC9250
- RTP over QUIC (RoQ) - draft-ietf-avtcore-rtp-over-quic
- Multipath QUIC - draft-ietf-quic-multipath

moq - Media over QUIC

MASQUE - Multiplexed Application Substrate over QUIC Encryption

Conclusiones

Conclusiones

1. QUIC es un protocolo de transporte como TCP o UDP
2. "Anti-osificación" (extensible)
3. Seguro (*)
4. Baja latencia en establecimiento de conexión
5. En desarrollo continuo

QUIC es una solución "anti-osificación", pero no el fin de la misma. TCP/UDP siguen ahí.

Q&A

written by human
not by AI

¡¡ Muchas gracias !!

https://nopacketloss.es/osificacion-de-internet-el-protocolo-quic/

oalfageme@nopacketloss.es

@oalfageme

Referencias

- QUIC 101
  https://www.youtube.com/watch?v=dQ5AND4DPyU&list=PLW_J3qpRhOzFilBV_Gt_C6pJLMNIDCPhP&index=72&t=1419s

- QUIC Will it replace TCP/IP?
  https://www.youtube.com/watch?v=A7NbvlswQks&list=PLW_J3qpRhOzFilBV_Gt_C6pJLMNIDCPhP&index=70&t=3223s

- Explaining QUIC
  https://www.youtube.com/watch?v=sULCOKfc87Y&list=PLW_J3qpRhOzFilBV_Gt_C6pJLMNIDCPhP&index=71

- Will QUIC kill TCP?
  https://www.youtube.com/watch?v=0B00TQ14faw&list=PLW_J3qpRhOzFilBV_Gt_C6pJLMNIDCPhP&index=69

Referencias (II)

- QUIC Protocol Tutorial https://www.youtube.com/watch?v=31J8PoLW9iM&list=PLW_J3qpRhOzFilBV_Gt_C6pJLMNIDCPhP&index=67&t=5248s

- IETF QUIC v1 Design https://www.cse.wustl.edu/~jain/cse570-21/ftp/quic/index.html#:~:text=2.1%20QUIC%20Header,-Figure%203%3AQUIC&text=QUIC%20has%20two%20different%20types,after%20the%20first%20connection%20established.

- A quick look at QUIC https://2023.apricot.net/assets/files/APPS314/2023-03-01-quic-apri_1677636425.pdf

- HTTP/3 from A to Z https://www.smashingmagazine.com/2021/08/http3-core-concepts-part1/

- HTTP/3 Performance Improvements https://www.smashingmagazine.com/2021/08/http3-performance-improvements-part2/