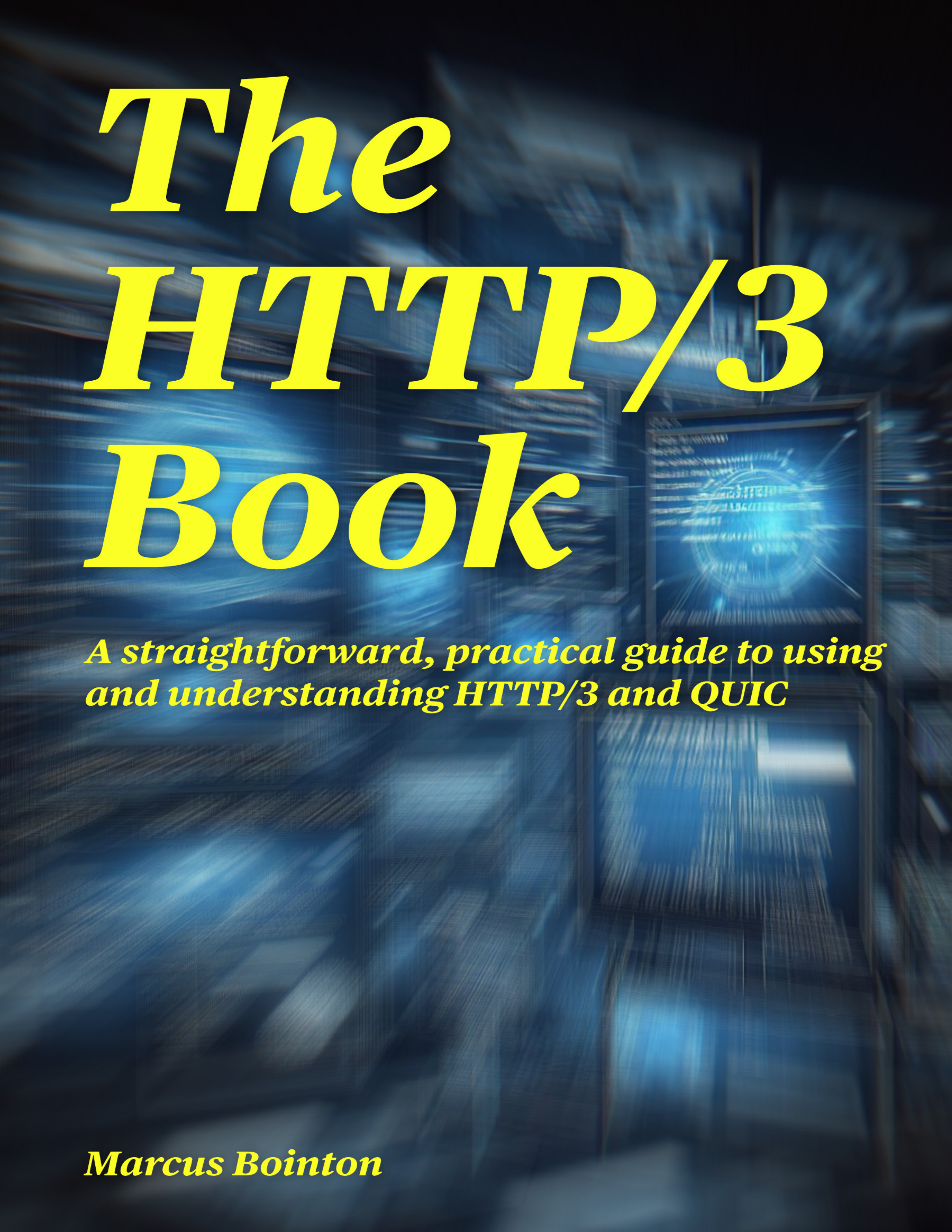

The HTTP/3 Book

A straightforward, practical guide to using and understanding HTTP/3 and QUIC

Marcus Bointon

The HTTP/3 Book

A straightforward, practical guide to using and understanding HTTP/3 and QUIC

Marcus Bointon

This book is available at https://leanpub.com/thehttp3book

This version was published on 2025-06-14

This is a Leanpub book. Leanpub empowers authors and publishers with the Lean Publishing process. Lean Publishing is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

Contents

Introduction

Welcome to The HTTP/3 book, a straightforward introduction to HTTP/3 and QUIC, the latest generation of web protocols.

This book offers an easy-to-understand overview of HTTP/3, the latest version of the hypertext transfer protocol, and QUIC, the new transport protocol designed to enhance the speed, security, and reliability of internet connections. As replacements for HTTP/2 and TCP, these protocols aim to solve persistent issues and bring useful improvements to web performance, particularly for those that need it most.

Whether you're a web developer, a networking beginner, or simply curious about how the web works, this guide provides a clear and concise explanation of these modern technologies.

The idea is not to overwhelm you with extreme detail, or provide an exhaustive reference – there is too much to cover in the stack of RFCs that make up the entirety of HTTP/3 and QUIC, and you don't need to know most of those details if your primary objective is just to get your site served securely and reliably over HTTP/3.

We'll take a look at how we got to this point, why these new protocols were needed, explore the basics of HTTP/3 and QUIC, and how to deploy and test your configuration. By the end, you'll have a solid foundation to understand these new technologies, their impact on the web, and know how to use them to deliver your own web services.

About the author

Hello! I'm Marcus Bointon[1], a skier, songwriter, software developer, entrepreneur, and privacy advocate. My journey with computers began in the 1980s, through two computing degrees, numerous support roles (including for HP and Apple), and I've been a professional developer since the 1990s.

Currently, I'm engaged in writing and penetration testing for Radically Open Security[2], developing clubzero.co[3] and smartmessages.net[4], and supporting 1CRM[5] and RUSI[6] in the UK.

Figure 1. A portrait of Marcus Bointon wearing a blue ski helmet and red/go

I'm a contributor to numerous open-source projects, including WordPress[7]and Laravel[8], but I'm mostly known as the maintainer of PHPMailer[9], the second-most forked PHP project on GitHub, and likely the world's most popular email sending code; as a bonus, I wrote the HTML5 email address spec.

I have strong skills in PHP coding, Linux system administration, technical writing, and MySQL admin. I'm a passionate advocate for digital rights and privacy, including GDPR. Since 2006, I've been speaking at technical conferences around the world. Electronic music is something of a passion, and I write songs about open source[10]! Though originally British, I'm now French, and I live in the French alps with my wife, kids, cat, guitars, bikes, and perhaps too many skis.

This is my first book, so I hope you like it! I'm always happy to hear from readers, so please feel free to get in touch with me through any of the channels below.

[1] https://marcus.bointon.com/
[2] https://radicallyopensecurity.com/
[3] https://clubzero.co
[4] https://info.smartmessages.net
[5] https://www.syniah.com
[6] https://rusi.org
[7] https://wordpress.org
[8] https://laravel.com
[9] https://github.com/PHPMailer/PHPMailer
[10] https://marcus.bointon.com/tag/song/

Channel	Link
My blog	marcus.bointon.com[11]
Mastodon	@Synchro@phpc.social[12]
Twitter/X	@SynchroM[13]
GitHub	Synchro[14]
LinkedIn	Marcus Bointon[15]
Stack Overflow	Synchro[16]
My music	Bandcamp[17]
Email	books@devalps.eu

Terminology

In case you're not familiar with some of the terms used in this book, here's a little glossary.

Bandwidth

The maximum rate at which data can be transferred over a network connection, usually measured in bits per second. See also throughput.

The proportion of available throughput used by a network operation. For example a file download would use a lot of bandwidth, and a DNS lookup very little.

Certificate

A digital document that is used to prove the identity of a server or client via cryptographic means. Certificates are used to establish secure connections over the internet, for example using TLS.

Cipher

An algorithm used to encrypt and decrypt data. Ciphers are used to secure data in transit over a network. The combination of a cipher and signature algorithm is called a cipher suite.

Client

A program that requests resources from a server using HTTP, for example a web browser like Safari or Chrome, or a mobile app.

Compression

A mathematical transformation used to reduce the size of data for transmission or storage, which can improve throughput and reduce bandwidth usage. Common compression algorithms include zip, gzip, zstd, and brotli.

DNS

Domain Name System, a system that translates human-readable domain names (like `example.com`) into IP addresses.

Firewall

A security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules.

Fork

A copy of a software project that is developed independently of the original project. Forks are often created when developers disagree about the direction of a project or when they want to experiment with new features.

An operation that creates a new process in an operating system.

Header

A block of data at the beginning of a packet that contains metadata about the packet, such as the source and destination addresses, the size of the packet, and the type of data it contains.

Metadata about the contents of an HTTP request or response.

Head-of-line blocking (HOLB)
A problem that occurs in some network protocols where one slow or lost packet can block the delivery of other packets that are behind it in the queue.

HTTP
HyperText Transfer Protocol, the protocol used for transferring web pages and other resources on the internet. HTTP usually involves discrete requests from clients that are responded to by servers. See How did we get here?

HTTPS
HyperText Transfer Protocol Secure, an extension of HTTP that uses TLS to encrypt and secure data in transit. HTTPS is used to secure connections over the internet, for example when you visit a website that starts with `https://`. HTTP versions 2 and 3 *always* use HTTPS.

IP Internet Protocol, a low-level protocol that is used to route data between computers on the internet.

A unique number (address) assigned to each device on a network that uses the Internet Protocol for communication.

IPv4
Internet Protocol version 4, the most widely used version of the Internet Protocol. Encodes addresses as a 32-bit number, giving a maximum of around 4 billion possible addresses, which is now significantly less than the total number of devices in use, and is thus subject to address exhaustion, i.e. running out of addresses.

IPv6
Internet Protocol version 6, the successor to IPv4, defined in 1998. Encodes addresses as a 128-bit number, giving a maximum of around 340 undecillion possible addresses (yes, this is a ludicrously huge number), which is more than enough to cover all devices on the internet.

Javascript
A programming language that is mostly used to create interactive applications that run inside web browsers. Often abbreviated to "JS".

Latency
The time it takes for a packet of data to travel from one point to another in a network. Often used to describe the time it takes for a request to reach a server from a client, or vice versa.

Library
A piece of software that (usually) does one specific thing. Applications often assemble multiple libraries into bigger, more useful contexts, so you might have a library that fetches things from a URL, another one that sends email, and use them to build an application that emails the contents of a URL.

Metadata
Data that describes other data. For example, the title of a book is metadata about the book; the number of words in this sentence is metadata about the sentence.

Network speed
Network speed is a combination of the latency and throughput of a network connection. A network with high throughput and low latency is considered "fast".

OSI model
The Open Systems Interconnection model, a conceptual framework that standardizes the

functions of a telecommunication or computing system into seven abstraction layers, from voltage on a wire all the way up to "order a pizza". The OSI model is often used to describe how different network protocols interact with each other. Now considered somewhat archaic, but still conceptually useful.

Packet

A unit of data that is transmitted over a network. Packets contain both the data being transmitted and a metadata header that describes the data, such as the source and destination addresses. A resource transferred over the internet is broken up into packet-sized chunks, sent, then reassembled in the right order at the receiving end. A packet is typically about 1,500 bytes in size.

Protocol

A set of rules that define how data is transmitted over a network. Protocols can define things like how data is formatted, transmitted, and received.

QUIC

Originally "Quick UDP Internet Connections", but now only ever referred to by its acronym, a new transport layer protocol built on top of UDP, developed by Google, and designed to replace separate TCP and TLS network layers.

Resource

A file or other data that is requested by a client from a server over the internet. Resources can include text, web pages, images, videos, database records, and other types of data.

Anything that has finite availability, such as processor time, computer memory, storage space, network bandwidth, power consumption.

RFC

Request for Comments, a publication from the Internet Engineering Task Force (IETF) that describes internet standards and protocols.

Round trip

An instance of data travelling from one network device to another and back again, for example an HTTP request and response, or a DNS lookup and reply.

The time it takes for a packet of data to travel from one point to another *and back again.*

Server

A program that provides resources to clients using a protocol, for example HTTP served by Apache or nginx.

A computer that runs server programs.

SPDY "Speedy"

A protocol developed by Google in 2010, now obsolete, that was the basis for HTTP/2.

TCP

Transmission Control Protocol, a reliable, connection-oriented protocol that is used to transfer data over the internet. TCP sits on top of IP, and the combination is often referred to as TCP/IP.

Throughput

The amount of data that can be transferred in a given amount of time, usually measured in bits per second.

TLS Transport Layer Security, a protocol that provides secure communication over a computer network. It replaced SSL in 1998. Commonly used as a wrapper around other protocols, such as HTTP, to provide encryption and authentication.

Transport

A layer in the OSI model that is responsible for delivering data between devices on a network. The transport layer is responsible for error-checking, flow control, and retransmission of lost data.

UDP

User Datagram Protocol, a connectionless protocol built on top of IP that is used to send data over the internet. Often used for "lossy" protocols like audio and video streaming, but also used for DNS. UDP is simpler and faster than TCP but does not provide the same reliability guarantees. If you liken TCP to making a phone call, UDP is like using a walkie-talkie.

URL

Uniform Resource Locator, a string that identifies a resource on the internet, for example `https://example.com`.

WebSockets

A protocol that provides full-duplex (both directions simultaneously) communication channels over a single TCP connection. WebSockets are often used to enable real-time communication between a client and a server. In contrast to HTTP's discrete requests and responses, WebSockets are characterised by a constant streams of data. HTTP servers can usually handle both kinds of traffic.

How did we get here?

Since the invention of the web in 1991, we've seen steady progress in the capabilities of the fundamental building blocks of the web: HTTP, HTML, and URLs.

HTTP has evolved through several versions, each introducing new features and improvements to address the changing needs of the web. Here's a brief overview of the major HTTP versions, their key features, and the RFCs that document them:

HTTP Version	Year	Key Features and Changes	Related RFCs
0.9	1991	Simple, line-based protocol. Only GET method supported. No headers or metadata.	N/A
1.0	1996	Added response status codes, headers, MIME, and additional methods (POST, HEAD).	RFC 1945[1]
1.1	1997	Host header, persistent connections, pipelining, chunked encoding, and new methods (PUT, DELETE, OPTIONS, TRACE, CONNECT). Improved caching and bandwidth utilization.	RFC 2068[2], RFC 2616[3], RFC 7230-7235[4] (obsoletes RFC 2616)
2.0	2015	Binary protocol, multiplexing, header compression, server push, and improved security.	RFC 7540[5], RFC 7541[6]
3.0	2022	HTTP semantics largely unchanged, but uses QUIC as the transport protocol instead of TCP.	RFC 9114[7]

Looking at the networking layer, there has only been one switch: from TCP to QUIC:

Protocol	Year	Key Features and Changes	Related RFCs
TCP	1974 (Initial spec)	Connection-oriented, reliable, ordered data transfer. Sliding window for flow control and congestion control algorithms.	RFC 793[8] (Initial Spec), RFC 5681[9] (TCP Congestion Control)

Protocol	Year	Key Features and Changes	Related RFCs
QUIC	2021 (Initial draft)	Connection-oriented, reliable, ordered data transfer designed for low-latency and improved security. Uses UDP as the base transport protocol. 0-RTT connection establishment, forward error correction, and integrated TLS 1.3 encryption.	RFC 9000[10] (QUIC Spec), RFC 8446[11] (TLS 1.3)

The first proposals for what would become QUIC were created by Jim Roskind at Google in 2012. The first IETF draft version of QUIC was released in 2015, but it wasn't until 2021 that the IETF published the first standardised version of the QUIC protocol as RFC 9000[12]. The combination of HTTP and QUIC became known as HTTP/3, and it was standardised in 2022 in RFC 9114[13].

[12] https://datatracker.ietf.org/doc/html/rfc9000
[13] https://datatracker.ietf.org/doc/html/rfc9114

What did HTTP/2 change?

HTTP/2 was a major revision of the HTTP protocol, and it introduced several key changes that have had a significant impact on the way the web works. HTTP/2's origins lie in a protocol called SPDY, developed by Google in 2010, and later adopted by the IETF. Vendors were quick to implement SPDY, but after HTTP/2 was finalised, they rapidly ditched it in favour of the newly ratified protocol.

Binary protocol

HTTP/2 switched to a refined form of the binary protocol introduced in SPDY, representing a major shift in HTTP's architecture which made several other options possible. While text-based protocols are easy for humans to work with, binary protocols can be more space-efficient and much faster to parse.

Header compression

Text-based protocols are not good for operations like compression and encryption, and the binary protocol allowed HTTP to enable compression of HTTP headers, not just the body, meaning more bang for your bits, and higher performance.

Multiplexing

HTTP/1.1 tried to improve the performance of parallel response delivery by using multiple parallel TCP connections (defaulting to 6 per domain in most browsers), but this also increased memory consumption and latency as each connection had to do a complete TCP and TLS handshake – this overhead is clearly visible in browser development tools (see the chapter on testing). Multiplexing allowed multiple resources to be transferred over the same TCP connection at the same time. This was a step up from the pipelining and keepalive introduced in HTTP/1.1 as it allowed dynamic rescheduling of resource delivery, allowing for example an important, but small, JSON response to sneak past a bigger, but less important image download, even if it was requested later.

Server push

Server push eliminates some requests, for example allowing multiple image or Javascript subresources to be speculatively bundled in the response to a single request for an initial HTML document. Note that despite the name, this is not an active push initiated by the server (like push notifications are), but a speculative pre-fill of the client's cache, which is then used to satisfy subsequent requests which might never happen.

TLS-only

During HTTP/2’s standardisation process there was a great deal of push-back from corporate interests to allow it to work over unencrypted channels, and that did ultimately make it into the RFC. However, implementers rejected the entire premise, and all popular implementations only support HTTP/2 over HTTPS, raising the security floor for everyone.

What problems does HTTP/2 have?

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Head of line blocking

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Network switching

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

It’s stuck with TCP

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

TCP congestion control

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Server push is unpopular

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

What are QUIC and HTTP/3?

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Network layers

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Connection improvements

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Head of line blocking

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Network switching

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Header compression

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Security upgrade

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Prioritisation

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HTTP/3 implementations

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HTTP/3 support in cURL

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HTTP/3 support in PHP

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HTTP/3 connections

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Initial connections

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Resumed connections

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Pipelining and multiplexing

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HTTP/1.1 pipelining

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HTTP/2 multiplexing

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HTTP/3 multiplexing

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Effects of multiplexing on compression

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

WebSockets over HTTP/3

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

WebTransport

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

QUIC networking

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

It's much like TCP

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Packet numbering

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Round trip time (RTT) measurement

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Packet loss recovery

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Congestion control

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

TLS integration

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Deploying HTTP/3

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Alt-Svc DNS records

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

SVCB DNS records

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Which method should you choose?

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Deploying HTTP/3 via CDNs

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

CloudFlare

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Fastly

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Configuration for HTTP/3

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Caddy

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Nginx

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Microsoft IIS

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Apache

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HAProxy

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Firewalls

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Firewall issues

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Cloud provider security groups

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Optimising for HTTP/3

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Don't bundle resources

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Use fewer domains

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Use `async`, `defer`, and `preload`

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Use `rel=preconnect` and `rel=dns-prefetch`

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Apply UDP optimisations

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Testing HTTP/3

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Testing tools

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HTTP version test site

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Waterfall analysis

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HTTP/3 problems

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

The future of QUIC

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

HTTP/3 and QUIC resources

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

RFCs

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Articles and blog posts

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.

Tools

This content is not available in the sample book. The book can be purchased on Leanpub at https://leanpub.com/thehttp3book.