University of Zurich^UZH

MASTERS THESIS – Communication Systems Group, Prof. Dr. Burkhard Stiller

Measuring QUIC Censorship in Understudied Regions

Mohamed Zahir Mohamed Wazeer
Zürich, Switzerland
Student ID: 21-742-150

Supervisor: Thomas Grübl, Daria Schumm
Date of Submission: October 15, 2025

University of Zurich
Department of Informatics (IFI)
Binzmühlestrasse 14, CH-8050 Zürich, Switzerland

ifi

Master Thesis
Communication Systems Group (CSG)
Department of Informatics (IFI)
University of Zurich
Binzmühlestrasse 14, CH-8050 Zürich, Switzerland
URL: http://www.csg.uzh.ch/

Declaration of Independence

I hereby declare that I have composed this work independently and Generative AI was used to improve the coherence of sentences and paragraphs. I am aware that I take full responsibility for the scientific character of the submitted text myself, even if AI aids were used and declared (after written confirmation by the supervising professor). All passages taken verbatim or in sense from published or unpublished writings are identified as such. The work has not yet been submitted in the same or similar form or in excerpts as part of another examination.

Zürich, 14 October 2025

Signature of student

Zusammenfassung

Diese Arbeit untersucht die Internetzensur auf Anwendungsebene, die auf das QUIC-Protokoll abzielt. Dabei liegt der Schwerpunkt darauf, wie bestimmte Netzwerke den QUIC-Verkehr stören und gleichzeitig die traditionelle TCP/TLS-Kommunikation ermöglichen. Nach einer umfassenden Literaturrecherche zu Internetzensurtechniken und den Designmerkmalen von QUIC wurde eine Liste verifizierter QUIC-unterstützter Domänen erstellt, wobei die Schweizer Perspektive als zensurfreie Basis diente. Anschließend wurde mithilfe der aioquic-Bibliothek ein modulares Prüftool entwickelt, um kontrollierte QUIC- und TCP/TLS-Handshakes durchzuführen. Das Tool wurde auf mehreren Virtual Private Servern (VPS) in Süd- und Südostasien eingesetzt, um regionsspezifische Blockierungsmuster zu erkennen. Jede Prüfung zeichnete detaillierte Handshake-Ergebnisse, Erfolgscodes und Fehlerspuren auf, um Fälle zu identifizieren, in denen QUIC-Verbindungen fehlschlugen, TCP jedoch erfolgreich war, was auf potenzielle Störungen auf Anwendungsebene hindeutet. Die analysierten Ergebnisse zeigen klare Hinweise auf QUIC-spezifische Blockierungen in einigen Netzwerken, während andere eine normale Konnektivität oder allgemeine Störungen auf Netzwerkebene aufwiesen. Die Studie kommt zu dem Schluss, dass QUIC zwar in den meisten getesteten Regionen weiterhin zugänglich ist, die selektive Filterung des QUIC-Verkehrs jedoch weiterhin besteht. Dies verdeutlicht die Entwicklung von Zensurpraktiken im Zuge der zunehmenden Verbreitung verschlüsselter, UDP-basierter Protokolle.

Abstract

This thesis examines application-layer Internet censorship targeting the QUIC protocol, focusing on how certain networks interfere with QUIC traffic while allowing traditional TCP/TLS communication. After conducting an extensive literature review on Internet censorship techniques and QUIC's design characteristics, a list of verified QUIC-supported domains was compiled using a Swiss vantage point as a censorship-free baseline. A modular probing tool was then developed using the aioquic library to perform controlled QUIC and TCP/TLS handshakes. The tool was deployed across multiple Virtual Private Servers (VPSs) located in South and Southeast Asia to detect region-specific blocking. Each probe recorded detailed handshake outcomes, success codes, and error traces to identify cases where QUIC connections failed but TCP succeeded, indicating potential application-layer interference. The analyzed results show clear evidence of QUIC-specific blocking in some networks, whereas others demonstrated normal connectivity or general network-layer disruptions. The study concludes that while QUIC remains accessible in most tested regions, selective filtering of QUIC traffic persists, highlighting evolving censorship practices as encrypted, UDP-based protocols become more widespread.

Acknowledgments

I would like to express my deepest gratitude to my supervisor, Thomas Grübl, for his continuous support and guidance throughout the course of this thesis. His constructive criticism, thoughtful suggestions, and material support provided at the outset were invaluable in shaping my understanding and approach to the research. I am especially grateful for his commitment and availability, even during his holidays, which reflects his exceptional dedication as a mentor.

A warm thanks also goes to the entire Communication Systems Group, led by Professor Burkhard Stiller, for making this thesis possible. The group has played a central role in my academic journey during the master's program, where I had the privilege of engaging with almost all its members through various courses. Finally, I extend my heartfelt appreciation to my family for their patience and understanding, especially for bearing with the reduced time I could spend with them during this demanding period.

Contents

Chapter 1

Introduction

Internet censorship, in simple terms, refers to the restriction or denial of access to the internet or specific websites for users. These restrictions are often driven by a variety of factors, most notably geopolitical motivations. Censorship presents a persistent challenge that undermines the right to free expression and open internet access. Governments employ increasingly sophisticated techniques to block external traffic at the national network perimeter using censorship devices. Common methods include Deep Packet Inspection (DPI), Domain Name System (DNS) manipulation, and Internet Protocol (IP) address blocking [1].

While traditional censorship mechanisms have predominantly targeted protocols such as HTTP and HTTPS over TCP, the advent of new transport protocols like QUIC introduces fresh challenges. Unlike TCP, QUIC encrypts more of its metadata, making it difficult for censorship tools to analyze packet contents. Developed initially by Google and standardized by the IETF [2] QUIC enhances web performance through faster connection establishment and improved security by integrating TLS encryption from the initial handshake. By operating over UDP and encrypting transport-layer information, QUIC reduces the visibility that censors typically rely on to detect and block traffic.

Censorship is particularly prominent in regions such as Eastern Europe, Russia, and China, where access to global content is heavily filtered. The “Great Firewall of China” is a notable example, known for its highly fortified and complex censorship infrastructure [3, 4]. However, many countries in Asia, Latin America, and Africa remain understudied, and little is documented about their censorship practices, especially in relation to QUIC, which is still an emerging protocol compared to the widely deployed TCP.

This thesis aims to investigate the censorship of the QUIC protocol in a selected understudied region, thereby contributing to the body of knowledge concerning the effectiveness and impact of censorship strategies on modern web protocols.

1.1 Motivation

Internet censorship encompasses the deliberate limitation or obstruction of access to online content or services, typically enforced by governments or institutional entities. These interventions are often motivated by political agendas, ideological control, or national security concerns, posing a significant threat to fundamental digital rights such as free expression and open access to information [5]. In recent years, censorship strategies have become more technologically advanced, with network-level enforcement mechanisms becoming increasingly prevalent. Among these are techniques such as Deep Packet Inspection (DPI), manipulation of the Domain Name System (DNS), and IP address filtering, all of which enable selective disruption or redirection of user traffic at scale [1].

Historically, such controls have primarily targeted widely used protocols like HTTP and HTTPS, both of which operate over the TCP transport protocol. However, the emergence of new protocols, most notably QUIC, has introduced new layers of complexity for censors, which was initially introduced by Google, runs over the UDP transport layer, thereby reducing protocol visibility and making conventional filtering less effective [2]. With TLS encryption integrated into its initial handshake and limited exposure of transport-layer details, QUIC diminishes opportunities for packet inspection but not eradicated. Consequently, censorship apparatuses are increasingly relying on protocol fingerprinting and traffic pattern analysis to detect and obstruct QUIC-based communication [8].

While heavily censored countries like China, Russia, and Iran are known for their sophisticated and expansive internet control frameworks, less attention has been given to how emerging protocols like QUIC are handled in other parts of the world. The "Great Firewall" of China remains a key case study in censorship research due to its scale and technical depth [1]. However, numerous nations across Asia, Latin America, and Africa remain understudied in this context. This thesis seeks to address that gap by examining the extent and nature of QUIC censorship in a selected understudied region, contributing to the broader discourse on evolving forms of internet regulation and their implications for digital freedoms.

1.2 Description of Work

This study provides an experimentally grounded evaluation of QUIC-specific censorship, demonstrating how selective interference can be detected and characterized across multiple regions. The results reveal that censorship targeting QUIC operates with precision, indicating a deliberate and protocol-aware filtering approach rather than widespread transport-layer suppression. The methodology applied in this research centered on empirical probing, packet-level validation, and controlled port variation proved effective for distinguishing genuine interference from transient network anomalies.

Looking ahead, future work could extend this analysis through systematic multi-port probing to understand the depth of protocol discrimination, URL-path fuzzing to assess potential content-based filtering, and longitudinal measurements to observe temporal or event-driven variations. Such continued investigations would contribute to a more comprehensive understanding of evolving censorship practices and their implications for encrypted transport protocols like QUIC.

1.3 Thesis Outline

The structure of this thesis is organized as follows. Chapter 2 presents the necessary background, discussing the mechanisms of Internet censorship and circumvention, along with an introduction to the QUIC protocol. Chapter 3 reviews related work, identifying existing research efforts, analyzing their methodologies, and highlighting their limitations and open challenges. Chapter 4 details the design and development of the measurement pipeline, describing the tool to be built for analyzing censorship targeting QUIC traffic. Chapter 5 focuses on the infrastructure setup, elaborating on the deployment of the tool and the integration of the necessary modules to facilitate real-world measurements. Chapter 6 presents the experiments and analysis of the results collected during the measurement campaigns. Finally, Chapter 7 concludes the thesis by summarizing the findings, discussing limitations, and proposing directions for future research.

Chapter 2

Background

This section provides the foundational background for analyzing how the QUIC protocol is subjected to censorship, particularly in less documented regions. Internet censorship can be broadly defined as the deliberate suppression or restriction of online content or services by governmental or institutional entities, often motivated by political, social, or security concerns [1, 5]. Common censorship strategies include blocking IP addresses, tampering with DNS responses, and performing content-based filtering through Deep Packet Inspection (DPI) or keyword detection [7, 5].

The emergence of QUIC has introduced new challenges for censorship systems [2]. Unlike traditional TCP based protocols such as HTTP/1.1 and HTTP/2, QUIC uses HTTP/3 which encrypts not only the application data but also most of its transport layer metadata, including connection establishment details. This encryption reduces protocol visibility and impairs traditional censorship tools that rely on header inspection or static port-based filtering [6].

While censorship mechanisms have been extensively studied in high surveillance environments such as China and Iran, research on the censorship of modern protocols in other parts of the world especially in regions across Asia, Africa, and Latin America remains limited [8]. These regions may implement different technical or policy-driven censorship models that are less visible but equally impactful. Investigating the status of QUIC in such contexts is essential to expanding our understanding of global censorship dynamics. This chapter sets the stage for such an inquiry by situating the research within the broader evolution of censorship technologies and emphasizing the significance of underexplored geographies in modern network regulation.

2.1 An Overview of Censorship

When a user attempts to access fictitious websites such as xyz.com, def.com, or abc.com, the middlebox intercepts the traffic, inspects it, and applies predefined filtering rules to determine accessibility. As illustrated in Figure 2.1, the middlebox allows the user to

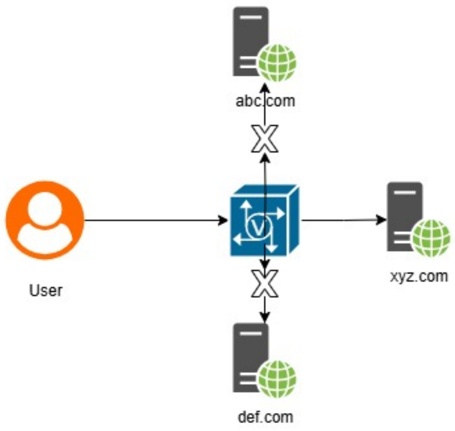


Figure 2.1: Illustration of Internet Censorship Through Traffic Filtering.

successfully reach xyz.com, while access to def.com and abc.com is denied, demonstrating its role in selectively controlling connections between the user and the web servers.

Internet censorship refers to the intentional control, suppression, or restriction of access to online information, services, or platforms by governing bodies, private entities, or network operators [9]. This form of information regulation may involve technical interventions such as IP blocking, DNS manipulation, content filtering, or deep packet inspection [1], and is often enacted to enforce political, social, cultural, or security driven objectives [10]. While sometimes justified as a means of national security, misinformation control, or legal compliance, internet censorship is frequently criticized for undermining fundamental rights such as freedom of expression, access to information, and digital privacy [11]. The extent and methods of censorship vary significantly across geopolitical contexts, ranging from selective filtering to comprehensive nationwide firewalls [1].

2.1.1 Purposes Commonly Associated with Censorship

The objectives of internet censorship are shaped by a variety of political, social, and economic factors, often reflecting the priorities and governance style of a state [12]. In autocratic or semiauthoritarian systems, censorship is frequently utilized to suppress political dissent, limit the spread of oppositional narratives, and maintain regime stability [10]. In contrast, culturally or religiously conservative societies may implement restrictions to prevent access to content deemed morally inappropriate or socially disruptive [13]. States may also justify censorship on national security grounds, particularly during emergencies or crises. For example, following the Easter Sunday bombings in 2019, the Sri Lankan government imposed a temporary ban on major social media platforms in an effort to prevent the circulation of false information and inflammatory content that could incite further unrest [15]. Although such actions are often positioned as precautionary measures, they simultaneously raise concerns about overreach and the erosion of civil liberties [10]. Even in democratic contexts, limited censorship may be applied to protect children from harmful material, enforce copyright laws, or comply with judicial rulings [14]. Additionally, economic considerations may lead governments to block foreign platforms or services to promote local industries. Overall, censorship operates as a multi-faceted tool used to manage digital communication, assert regulatory control, and shape the flow of information within and across national borders.

2.1.2 Implications for Digital Privacy and Access

The implementation of internet censorship carries profound consequences for digital rights, particularly in the areas of user privacy, the freedom to express ideas, and equitable access to knowledge. Surveillance mechanisms embedded within censorship infrastructures often permit the tracking of online activity, thereby diminishing the confidentiality of personal communications and undermining user trust in digital platforms [16]. This intrusion into private spaces fosters an atmosphere of fear and compliance, where individuals are discouraged from engaging in open or critical discourse. In societies with stringent information controls, state actors frequently suppress political dissent by restricting or disabling access to independent journalism, social media, and platforms that facilitate public discussion [10]. Such limitations disrupt the free flow of ideas and hinder the capacity of individuals to participate in meaningful societal debates. Moreover, censorship introduces barriers to information access, especially when educational, scientific, or internationally sourced content is filtered or made inaccessible [17]. As a result, affected populations may experience a narrowing of perspective and reduced exposure to global developments, contributing to social and intellectual isolation. These effects collectively weaken democratic resilience, marginalize diverse voices, and foster unequal access to digital resources.

2.2 Mechanisms for Enforcing Digital Censorship

Internet censorship can occur at various levels of the network stack, but the most prevalent forms target the transport and application layers. Transport-layer censorship operates by blocking or interfering with the basic means of communication such as TCP and UDP traffic, IP addresses, or port numbers, often through simple mechanisms like IP blocking or dropping UDP packets to prevent access to protocols like QUIC or DNS [22, 42]. While effective, this approach is typically broad and blunt, risking overblocking of legitimate services that share the same technical characteristics. In contrast, application-layer censorship is more granular and targeted, relying on the inspection of protocol-specific metadata and content to selectively disrupt access to certain services or websites. Common techniques include SNI filtering, where censors examine the unencrypted Server Name Indication during a TLS handshake to block specific HTTPS domains [25, 43]. HTTP Host header inspection, which allows filtering based on domain names in plaintext HTTP requests [44], and keyword-based filtering, where packet inspection tools monitor web traffic for politically sensitive or prohibited terms, often injecting TCP reset packets when matches are found [45]. These application-layer methods allow for more precise control and are increasingly used in countries with sophisticated censorship infrastructure. Because application-layer protocols often expose human-readable information, they offer state-level actors the ability to identify, classify, and block specific online content without disrupting the entire transport channel, making them a preferred strategy in environments where political control over information is critical.

2.2.1 IP Address Censorship

IP Address Blocking is one of the most fundamental and widely used techniques for implementing internet censorship. In this method, access to a specific server or service is denied based on its IP address, effectively cutting off all communication between clients and the targeted server. Governments or network operators configure routers or firewalls to drop or reject packets destined for these IP addresses [1]. This approach impacts all traffic, including QUIC, HTTP/3, TCP, and others, as long as it shares the same endpoint. While IP blocking is relatively easy to implement and effective against static services, it becomes less reliable in the face of content delivery networks (CDNs), domain fronting, or technologies like QUIC that employ connection migration and encrypted transport layers [18]. In the context of understudied regions, where infrastructure limitations or political constraints shape censorship strategies, measuring IP based blocking provides valuable insights into the scope and sophistication of censorship practices, particularly when alternative access methods are restricted or unavailable [19].

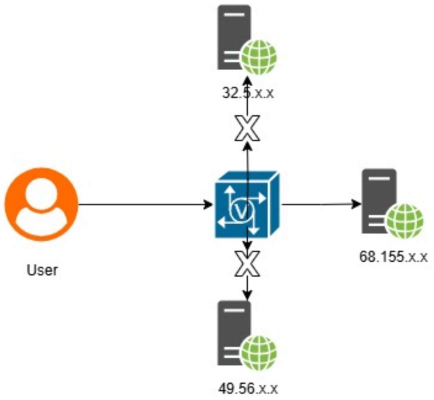


Figure 2.2: Network Layer Censorship

As illustrated in Figure 2.2, the middlebox enforces filtering at the network layer by inspecting the destination IP address of outgoing traffic. In this fictitious example, only the web server with the address 68.155.x.x is permitted to establish a connection and deliver content back to the user, while requests to other IP addresses are blocked outright. This form of control demonstrates that censorship can occur independently of the application or website being hosted, since access is denied based solely on the IP identifier rather than the content or service it provides.

2.2.2 Censorship by DNS Manipulation

Censorship through Domain Name System manipulation and hijacking targets one of the foundational layers of internet functionality, by distorting the process that maps domain names to their corresponding IP addresses. This type of interference allows state level or ISP level actors to prevent access to websites without blocking the underlying IP addresses directly. For instance, users attempting to visit a censored website may be redirected to a false or unrelated IP address, served a fake "site not found" error, or shown a government issued warning page, all without their knowledge of the tampering [1]. These practices exploit the inherently insecure nature of conventional DNS, which was not designed with authentication or encryption in mind. As such, DNS responses can be easily forged or intercepted in transit, especially in regions lacking deployment of secure alternatives like DNSSEC or encrypted DNS protocols [18]. DNS based censorship is favored in many countries due to its low cost, scalability, and relative ease of implementation, especially in environments where central control over ISPs is strong [20]. Investigating these DNS layer tactics is essential for understanding the subtler, often invisible forms of censorship that can bypass user awareness and reduce the effectiveness of circumvention tools.

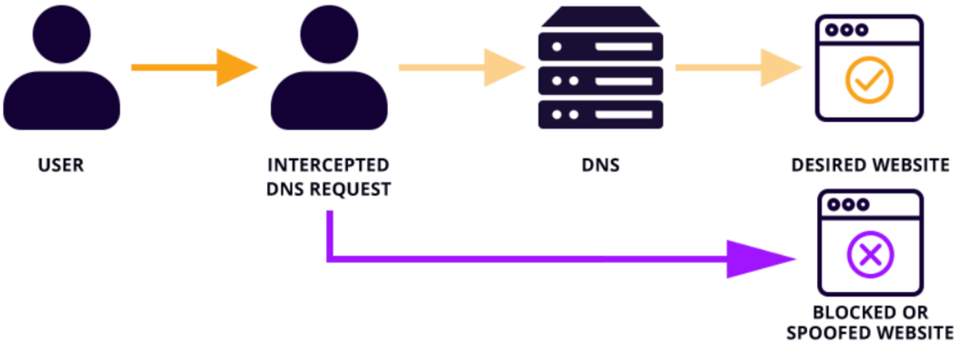


Figure 2.3: DNS Manipulation Reproduced from [21]

Figure 2.3 illustrates how a user's DNS request can be intercepted and altered before reaching the legitimate DNS server. In this process, the attacker or intermediary system manipulates the DNS entry and redirects the request to a false or blocked destination instead of the intended website. As a result, the user is either prevented from accessing the legitimate resource or unknowingly connected to a spoofed domain, demonstrating how DNS-level interference can be used as a censorship or redirection technique.

2.2.3 Deep Packet Inspection

Deep Packet Inspection (DPI) represents a sophisticated tool for implementing internet censorship, allowing realtime, in-depth analysis of data traffic. Unlike traditional filtering techniques that examine only packet headers, DPI scrutinizes the entire packet payload, enabling authorities to identify particular applications, communication protocols, or even specific keywords embedded in user activity [22].This finegrained visibility facilitates targeted interventions, such as blocking individual social media posts, detecting and disrupting anonymization tools like VPNs or Tor, and throttling encrypted protocols such as QUIC or HTTPS [23]. The adaptive capabilities of DPI make it more resilient than static censorship techniques, as it can dynamically adjust to changes in traffic patterns and circumvention strategies. As a result, many governments incorporate DPI into their broader censorship infrastructure to exert granular control over the flow of digital information while minimizing disruption to unrelated services [24]. However, its deployment raises substantial ethical concerns, as DPI also functions as a surveillance mechanism, undermining user privacy and freedom of expression.

2.2.4 Protocol Specific Blocking and Data Payload Inspection

Protocol based filtering enables censorship systems to restrict online access by targeting the underlying transport or application-layer protocols used in data transmission, rather than specific IP addresses or domains. This technique is especially relevant as users increasingly adopt encryption and circumvention tools. Censors identify and block protocols such as QUIC, HTTPS, Tor, and VPN tunneling by analyzing their traffic patterns, handshake structures, and behavioral signatures even when payloads are encrypted [22]. For instance, QUIC, designed to improve latency and security through multiplexed UDP based transport, has a distinct handshake and flow pattern that can be fingerprinted by Deep Packet Inspection (DPI) tools despite encryption [23]. Similarly, Tor traffic exhibits recognizable packet timing and size distributions, while VPN protocols like OpenVPN or IKEv2 can be flagged via port usage and handshake characteristics [24]. These protocol specific filters are often deployed in national firewalls or ISP-level infrastructure to enforce policy with greater precision. While effective, this approach is not foolproof users increasingly adopt protocol obfuscation and tunneling techniques (e.g., Tor pluggable transports, obfuscated QUIC) to disguise or randomize protocol traits, leading to an ongoing arms race between censorship and evasion technologies. Figure 2.4 depicts an example of QUIC protocol censorship within a national network. Internet Service Providers (ISPs) act as intermediaries that can filter or block traffic based on specific characteristics, such as Server Name Indication (SNI) values or QUIC version identifiers. When QUIC packets are detected, certain middleboxes or firewalls may terminate or drop these connections, preventing users from accessing the intended websites. This illustrates how censorship can occur at the transport or application layer through protocol-specific filtering mechanisms.

2.2.5 TCP Protocols as Early Targets of Internet Censorship

Censorship systems have traditionally concentrated on intercepting and restricting traffic that relies on TCP, particularly protocols such as HTTP and HTTPS, which are central to modern web communication. HTTP enables the retrieval of resources like web pages through plaintext exchanges over TCP connections, making it highly vulnerable to surveillance and blocking. Entities implementing censorship can inspect the contents of HTTP requests including URLs, headers, and payloads to identify and filter undesirable content

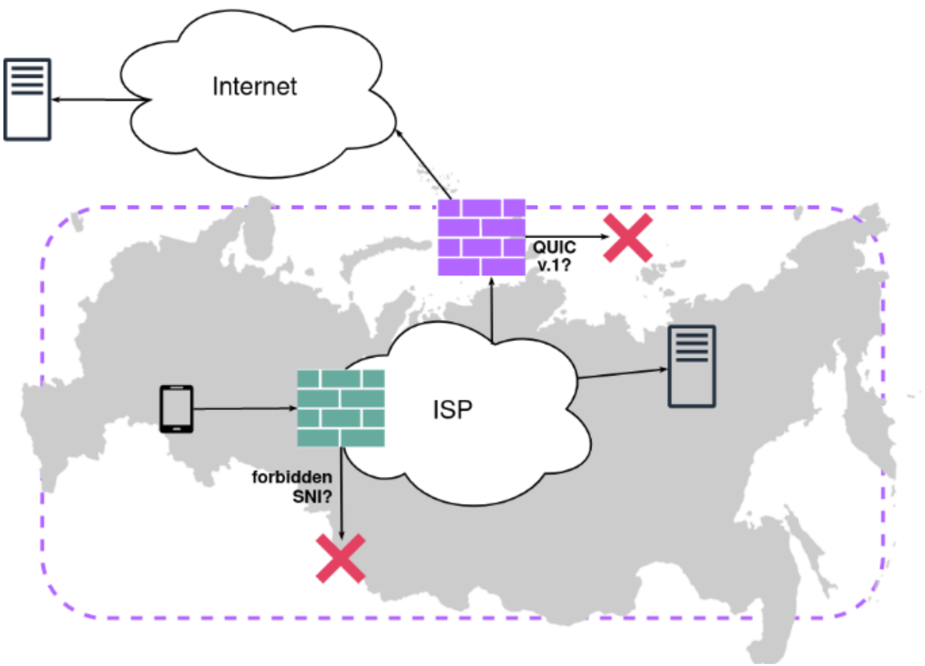


Figure 2.4: QUIC Protocol Censorship [25]

[27]. In response to privacy concerns, HTTPS was introduced to secure web traffic via TLS encryption, protecting the transmitted data from direct observation. Nevertheless, HTTPS still exposes certain metadata during the handshake phase, notably through the Server Name Indication (SNI) field and this depends on which TLS version is used, which reveals the destination domain name [31]. This, along with techniques such as TLS fingerprinting and analysis of traffic patterns, enables censors to infer the type of service or website being accessed without decrypting the actual content. Furthermore, authorities often resort to port level blocking, targeting TCP port 80 (used by HTTP) and port 443 (used by HTTPS), to restrict access to web services at a broader level [22]. These strategies leverage the transparency and consistency of TCP based protocols, making them a primary target for censorship efforts.

2.3 Transport Protocols and Their Influence on Internet Censorship

Transport protocols form a critical layer in the end-to-end delivery of online services, and their characteristics can make them either targets or enablers of censorship. While application-level filtering often relies on inspecting content or domain names, transport protocols such as TCP and QUIC determine how traffic is initiated, maintained, and encrypted across networks. Because these protocols define connection establishment, handshake procedures, and error handling, they can be selectively disrupted without directly interfering with higher-layer applications.

2.3.1 Challenges in Filtering UDP Compared to TCP

In contrast to TCP, which provides structured, stateful communication that facilitates content filtering, UDP presents significant obstacles for traditional censorship systems. As a connectionless and stateless protocol, UDP does not establish a persistent session between endpoints, nor does it offer mechanisms for sequencing, acknowledgment, or retransmission [27]. This lack of structure complicates the task of monitoring and identifying flows, as each UDP packet is independent and may arrive out of order or without context. These characteristics are increasingly exploited by modern protocols such as QUIC, which use encryption and multiplexing over UDP to obscure protocol signatures and prevent inspection of handshake and payload data [31]. Furthermore, many UDP-based applications implement port randomization and traffic obfuscation, further reducing the effectiveness of traditional blocking techniques. Stream filtering of UDP traffic is often impractical, as it risks disrupting essential services like DNS, VoIP, or real-time media streaming, which also rely on UDP. As a result, censorship targeting UDP-based protocols requires more advanced techniques such as behavioral fingerprinting or active probing, which are costly, complex, and more prone to false positives [22].

2.4 Introduction to QUIC and HTTP/3

The evolution of modern Internet transport protocols has been driven by the need for improved efficiency, reduced latency, and enhanced security. Traditional protocols like TCP, while foundational, introduced inherent performance constraints, particularly when supporting encrypted and multiplexed web traffic. In response to these limitations, new protocols such as QUIC and HTTP/3 emerged to streamline data transmission and strengthen privacy protections. This section provides an overview of the origins, design principles, and technical advantages of QUIC and its integration into HTTP/3, illustrating how they redefine transport-layer communication in todayâs Internet architecture.

2.4.1 Origin of QUIC and Development

The development of QUIC originated from Google's effort to improve internet performance by addressing long standing inefficiencies in TCP-based communication. Introduced around 2012, QUIC was designed to operate over UDP, enabling faster connection setups, minimizing latency, and integrating features like encryption and multiplexing at the transport layer. Google initially deployed the protocol within its ecosystem across services such as Chrome and YouTube to evaluate its potential in reducing load times and improving overall user experience [2, 32, 34].

Recognizing its technical advantages and broad applicability, Google submitted QUIC to the Internet Engineering Task Force (IETF) in 2016, initiating a multiyear standardization process. During this period, the protocol was significantly revised to meet open standards for security, modularity, and global interoperability. These efforts resulted in the formal publication of RFC 9000 in May 2021, which defined IETF QUIC as a standardized, encrypted transport protocol built on UDP [2, 27]. Today, IETF QUIC forms the transport foundation for HTTP/3 and is supported by a growing number of browsers, content delivery networks, and cloud providers, signifying its widespread adoption in modern web infrastructure.

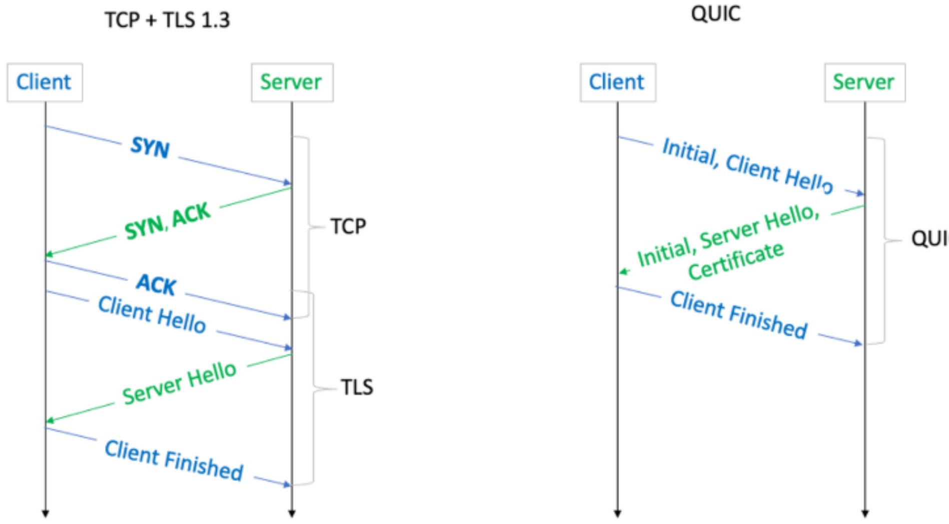


Figure 2.5: TCP Vs/ QUIC Hanshake [34]

Figure 2.5 illustrates the difference between the connection establishment process in TCP with TLS 1.3 and the QUIC protocol. In the traditional TCP and TLS handshake, multiple round trips are required to complete the connection setup and encryption negotiation. In contrast, QUIC integrates both transport and cryptographic handshakes into a single exchange, significantly reducing latency. This streamlined process enables faster connection establishment and improved performance for applications that rely on secure communication.

2.4.2 HTTP/2 Vs. HTTP/3

HTTP/2 was a major evolution from HTTP/1.1, introducing mechanisms like header compression, server push, and application layer multiplexing to enhance web performance. These improvements aimed to reduce latency and make better use of network resources. However, because HTTP/2 still relies on TCP as its transport layer, it inherits TCP's limitations most notably, head-of-line (HOL) blocking. Despite allowing multiple streams within a single connection, any packet loss at the TCP layer halts all streams until the missing data is retransmitted, resulting in avoidable delays [32].

To overcome this constraint, HTTP/3 replaces TCP with QUIC, a transport protocol developed on UDP. QUIC provides native stream multiplexing at the transport level, ensuring that packet loss in one stream does not interfere with others. It also integrates TLS 1.3 encryption directly into the transport protocol, eliminating the need for separate handshake layers and improving both security and connection speed. One of QUIC's key features, 0RTT resumption, further reduces latency by allowing repeat connections to start immediately without repeating the full handshake [2]. These architectural changes make HTTP/3 particularly effective in challenging network conditions such as mobile environments where it can offer faster, more reliable, and secure web communication compared to HTTP/2 [33].

QUIC leverages UDP to bypass the limitations of TCP, enabling faster and more flexible connection setups. It integrates TLS 1.3 encryption directly into the transport layer, reducing handshake latency and enhancing security without relying on separate layers. Additionally, QUIC supports stream level multiplexing, allowing multiple data streams within a single connection without interference. This design avoids head-of-line blocking, a common issue in TCP-based protocols like HTTP/2. Combined, these features make QUIC a secure, efficient, and performance oriented transport protocol for modern internet applications [18].

2.4.3 Privacy enhancements using QUIC

Following is a depiction of the constituency of a data packet of TCP and QUIC.

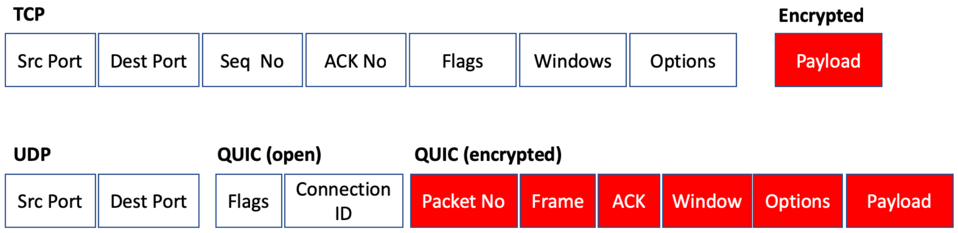


Figure 2.6: TCP vs. QUIC Data Packet Structure [34]

Figure 2.6 compares the data packet structures of TCP, UDP, and QUIC. While TCP packets include fields such as sequence numbers, acknowledgments, and window sizes, QUIC builds on UDP and introduces its own packet framing with elements like connection IDs, packet numbers, and frames. Unlike TCP, which provides encryption only when combined with TLS, QUIC inherently encrypts nearly all of its header fields and payload, enhancing both security and privacy. This design also enables QUIC to achieve faster and more flexible data transmission while maintaining encryption at the transport layer.

2.5 QUIC's Challenge to Traditional Censorship

The widespread adoption of QUIC has introduced major challenges for censorship systems built around TCPâs transparency. By encrypting transport-layer data, operating over UDP, and frequently changing connection identifiers, QUIC reduces the visibility that traditional filtering methods depend on. As a result, mechanisms such as Deep Packet

Inspection (DPI) and protocol fingerprinting struggle to detect or classify QUIC traffic accurately. The following subsections examine these limitations in DPI effectiveness and protocol identification.

2.5.1 Limitations of DPI Effectiveness in the Context of QUIC

QUIC significantly hinders Deep Packet Inspection (DPI) by encrypting most transport-layer data through its native integration of TLS 1.3. Unlike TCP, which exposes handshake and metadata elements used for filtering, QUIC conceals nearly all headers, including the Server Name Indication (SNI) [2, 25, 32]. Its variable-length headers and reduced retransmission timing further disrupt flow-based analysis, limiting network visibility for censorship systems [6]. Figure 2.6 illustrates how header encryption differs between TCP and QUIC.

2.5.2 Challenges in Protocol Identification

In contrast to TCP-based protocols, which benefit from standardized port assignments (such as port 443 for HTTPS) and uniform header formats that facilitate protocol identification, QUIC is built atop UDP and permits the use of arbitrary port numbers, thereby diminishing the reliability of port-based filtering techniques [2, 27]. Further complicating detection efforts, QUIC incorporates features such as variable-length headers, dynamic connection ID rotation, and frequent cryptographic key updates, all of which hinder attempts to derive consistent flow signatures for classification [6, 25]. Consequently, censorship systems struggle to accurately distinguish QUIC traffic from other benign UDP flows, often facing a dilemma between effective blocking and the inadvertent disruption of legitimate services tradeoff that undermines the granularity and effectiveness of protocol-level censorship [8, 22].

2.6 Insufficient Regional Coverage in Existing Internet Censorship Datasets

In summary, while existing censorship measurement initiatives have illuminated many aspects of state-level internet control, they remain limited by uneven geographic coverage. The concentration of data from a handful of well-studied countries, combined with infrastructural and political barriers in less-monitored regions, has resulted in significant blind spots [35, 8, 36]. This imbalance hinders the development of globally robust censorship circumvention strategies and risks overlooking localized techniques employed in under-represented areas. Recognizing and addressing these gaps is not only a methodological imperative but also a step toward a more equitable understanding of global information control. This thesis directly responds to this need by focusing on QUIC censorship in regions where empirical data remains sparse, thereby contributing to a more complete and representative view of contemporary internet censorship [37].

2.7 General Ethical Considerations

This research acknowledges the importance of conducting internet censorship measurements in a manner that is sensitive to the cultural, legal, and societal contexts of the regions involved specifically Thailand, Indonesia Malaysia, India, and Switzerland. In regions like Malaysia and India, where digital regulations and public discourse around online access vary greatly, careful consideration is given to ensure that measurement activities are non-intrusive and do not risk drawing unwanted attention to users or networks. In contrast, Switzerland, governed by strict privacy norms and GDPR-aligned data protection standards, demands strong safeguards around data handling and user anonymity. To respect these differences, the methodology avoids collecting personal data or interacting with user-specific content, and limits all probing to openly available public websites. The approach is designed to generate minimal network impact, with probing behavior that emulates ordinary user activity. By prioritizing transparency, restraint, and local legal compliance, this study aims to responsibly investigate censorship practices while upholding ethical standards across all involved jurisdictions.

Chapter 3

Related Work

This chapter reviews the body of research relevant to understanding and measuring QUIC censorship. It begins by examining published studies that have investigated the QUIC protocol and its interactions with censorship mechanisms, outlining the contributions and limitations of existing work. The second section explores the technical methods used by censors to detect and restrict internet traffic, focusing on how such techniques may apply to modern, encrypted protocols like QUIC. The third section assesses the capabilities and shortcomings of current censorship measurement platforms, particularly in relation to QUIC-specific detection. The final section highlights the scarcity of empirical censorship research in less-studied regions, underscoring the importance of extending measurement efforts to these areas, a gap this thesis seeks to address.

3.1 Existing Research on QUIC and Protocol Censorship

Table 3.1 lists key studies on QUIC and protocol censorship, outlining the main authors and research focus in this field. The resilience of QUIC against internet censorship has become a subject of growing academic and operational interest, especially as governments continue to refine their surveillance and blocking capabilities. Although QUIC was designed to obscure protocol metadata and reduce detectability through encryption and UDP-based transport, research shows that these protections are not foolproof. Bock et al. [6] demonstrated that Chinese censors have already developed techniques to identify and block QUIC connections by analyzing handshake characteristics and transport-layer patterns, despite the encryption of most protocol elements.

Complementing this, Elmenhorst [25] found that while QUIC is largely accessible in many countries, active interference remains prevalent in places like China and Iran. Broader studies into obfuscation and traffic fingerprinting, such as those by Fiore et al. [18] and Houmansadr et al. [24], further confirm that encrypted traffic can still be classified based on side-channel indicators like packet size distributions and handshake consistency. More recently, censorship involving QUIC has expanded into Eastern Europe and Central Asia. In Russia, OONI [38] reported that ISPs have begun targeting HTTP/3 over QUIC, using

Authors	**Year**	**Description**
Clayton et al. [1]	2006	Early analysis of GFW circumvention using packet manipulation techniques.
Crandall et al. [19]	2007	Tool for tracking censorship dynamics by observing concept-based blocking.
Park and Crandall [3]	2010	Backbone-level filtering analysis in China using national-scale IDS data.
Knockel et al. [23]	2011	Empirical study on China's censorship mechanisms using diverse datasets.
Weinberg et al. [36]	2011	Proposal for a global framework to monitor internet censorship behavior.
Aryan et al. [4]	2013	Initial look into internet censorship tactics employed by Iran.
Houmansadr et al. [24]	2013	Theoretical models for observing encrypted network communications.
Alam et al. [22]	2018	Comprehensive survey of censorship resistance strategies and protocols.
Jones et al. [8]	2020	Challenges in measuring censorship at scale using global data sources.
Fiore et al. [18]	2020	Study on obfuscation metrics and censorship resistance in encrypted protocols.
Sen et al. [38]	2020	Study of regional filtering patterns and their network performance impact.
Bock et al. [6]	2021	Empirical analysis of QUIC censorship in China using handshake fingerprinting.
GitHub/net4people [40]	2021	Documentation of HTTP/3 (QUIC) blocking behaviors by Russian ISPs.
Elmenhorst [26]	2022	Overview of global QUIC censorship, highlighting interference in China and Iran.
OONI [39]	2023	Measurement of QUIC and HTTP/3 blocking in Russia during military conflict.
OONI [41]	2023	OONI-based analysis of media and tool censorship in Azerbaijan.
Reporters Without Borders [42]	2023	RSF report on Kazakhstan's heavy internet filtering and TLS interception.

Table 3.1: Selected References on QUIC and Protocol Censorship

deep packet inspection and port filtering to disrupt access to international services. This is echoed by technical community findings which document the systematic blocking of UDP port 443, commonly used by QUIC, across several Russian networks [39]. Azerbaijan has also intensified its censorship landscape, particularly during politically sensitive periods, by restricting access to independent media and circumvention tools, a pattern that may extend to QUIC-based traffic [40]. Likewise, Kazakhstan has employed measures such as HTTPS interception and platform-level blocking, indicating that it possesses the infrastructural capacity to target protocols like QUIC [41]. Together, these studies highlight that encrypted transport protocols remain vulnerable in adversarial environments where state-level actors possess both the intent and the means to perform advanced traffic discrimination.

3.2 Censorship Mechanisms and Detection Methodologies

Internet censorship involves a range of technical strategies used by states and network operators to restrict access to online content. Common techniques include IP address blocking, DNS tampering, keyword filtering via deep packet inspection (DPI), TCP reset injections, and protocol-specific interference. Earlier studies have shown how these techniques operate at scale. For instance, Park and Crandall [3] documented backbone-level HTML filtering in China, while Aryan et al. [4] reported keyword filtering and content manipulation in Iran. More drastic interventions, such as full network shutdowns, have been observed during political events, as shown in Dainotti et al.'s analysis of nationwide outages [20, 42]. As censorship efforts adapt to encryption, newer protocols like HTTPS and QUIC are increasingly targeted using traffic fingerprinting, SNI analysis, and transport-layer protocol blocking [6, 25].

Table 3.2 lists selected studies on internet censorship mechanisms, highlighting key authors and their areas of investigation. Detection of such interference has evolved accordingly. Traditional methods include active probing from distributed vantage points and response anomaly analysis, used in platforms like OONI [38, 40] and tools like Concept Doppler [19]. Fiore et al. [18, 43] and Houmansadr et al. [24] introduced obfuscation-aware measurements and differential traffic analysis to distinguish censor-induced anomalies from benign network behavior. As the landscape grows more complex, researchers also emphasize ethical challenges in high-risk regions and the need for careful deployment [8, 35]. QUIC, which operates over UDP and encrypts most metadata, presents unique challenges to detection systems that previously relied on observable handshake behavior and packet headers.

Recent efforts to measure QUIC-specific censorship have employed protocol-aware probing and comparison-based experiments. Bock et al. [6] used QUIC Initial packets to detect blocking in China by testing handshake completion across Autonomous Systems. Elmenhorst [25] conducted measurements comparing QUIC (HTTP/3) and HTTP/1.1 responses to uncover selective blocking in China and Russia. GitHub reports have similarly identified cases where ISPs block UDP port 443 while allowing TCP 443 [39]. Mishra et al. [32] analyzed HTTP/3 performance under varying network conditions, indirectly revealing infrastructure limitations or censorship. In parallel, Wendzel et al. [46] surveyed censorship measurement techniques, highlighting gaps in QUIC-related research and stressing the need for protocol-specific methodologies. These works reflect a growing shift toward nuanced and adaptive censorship detection as network protocols continue to evolve.

Authors	**Year**	**Description of Work**
Crandall et al. [19]	2007	Introduced ConceptDoppler to monitor censorship patterns.
Park and Crandall [3]	2010	Detected HTML content filtering in China's backbone.
Weinberg et al. [36]	2011	Promoted the idea of a global censorship observatory.
Dainotti et al. [20]	2011	Investigated internet shutdowns during political unrest.
Aryan et al. [4]	2013	Revealed DPI-based keyword filtering in Iran.
Houmansadr et al. [24]	2013	Investigated unobservable network communications.
Jones et al. [8]	2020	Described challenges in measuring large-scale censorship.
Fiore et al. [18]	2020	Proposed obfuscation metrics for censorship-resistant protocols.
Fiore et al. [44]	2020	Extended prior work on detecting obfuscation.
Bock et al. [6]	2021	Used QUIC probes to detect blocking in China.
Mishra et al. [33]	2021	Evaluated HTTP/3 performance across network types.
GitHub/net4 [40]	2021	Noted QUIC being blocked via UDP port 443 in Russia.
Elmenhorst [26]	2022	Compared QUIC and HTTP/1.1 for signs of selective blocking.
OONI [39]	2023	Reported increased censorship in Russia post-Ukraine war.
OONI [41]	2023	Documented persistent censorship in Azerbaijan.
Wendzel et al. [48]	2025	Surveyed modern censorship detection methods.

Table 3.2: Selected References on Censorship Mechanisms

3.3 Limitations of Current Measurement Frameworks

With the emergence of protocols like QUIC, existing internet censorship measurement platforms face growing challenges in accurately detecting interference. Legacy tools such as OONI Probe, ConceptDoppler, and ICLab [8, 19, 38] were originally designed for traditional, unencrypted protocols like HTTP and DNS, which exposed much of their handshake and payload information for inspection . These platforms rely on observable features such as plaintext headers, consistent packet patterns, or unencrypted Server Name Indication (SNI) fields to infer censorship. However, QUIC encrypts most of its handshake and metadata, and runs over UDP, making such inspection techniques far less effective. For instance, failures in QUIC connections may go undetected or be wrongly attributed to normal packet loss or server-side issues rather than active censorship [6, 25]. Elmenhorst et al. [47] highlighted this issue in their comparative study of HTTP/3 (QUIC) and HTTP/1.1 accessibility across several regions, showing that traditional probes often miss selective blocking of HTTP/3, particularly in countries like China and Russia, due to the limited visibility offered by encrypted QUIC connections.

Beyond these technical issues, current frameworks also lack comprehensive geographic reach and adaptability. Much of the existing measurement infrastructure is concentrated in countries with active research communities, while areas such as Central Asia, sub-Saharan Africa, and the Caucasus remain underrepresented in large-scale studies [35, 40, 41]. This leaves important gaps in global censorship monitoring. Moreover, some states have adopted advanced techniques such as dynamic port-based filtering, protocol-specific blocking, or deep packet inspection to interfere with HTTP/3 traffic over UDP 443 [32, 39]. These tactics often escape detection by tools that are not QUIC-aware or regionally deployed. Longitudinal studies are also rarely conducted, even though censorship behavior may change significantly during elections, protests, or geopolitical crises. As emphasized by Wendzel et al. [46], there is a pressing need for frameworks that are not only capable of handling encrypted protocols but also flexible enough to detect nuanced and time-sensitive censorship strategies in diverse political environments .

3.4 Regional Gaps in Censorship Studies

While QUIC censorship has been studied in depth in countries with established research infrastructures such as China, Russia, and Iran [6, 25, 38], many other regions remain significantly underrepresented. Areas like India, Malaysia, Thailand and much of Central Asia and sub-Saharan Africa lack consistent measurement coverage.[40, 41, 47]. This geographic imbalance restricts our global understanding of protocol-specific interference, particularly in politically dynamic regions where censorship may be evolving rapidly. As emphasized by Wendzel et al. [46], current frameworks often fail to detect subtle, region-specific interference strategies highlighting the urgent need for scalable, QUIC-aware measurement systems that can operate across diverse and underserved regions.

This thesis contributes toward addressing this gap by focusing on the detection of QUIC censorship in several of these non-exposed or understudied countries, thereby extending the reach of empirical censorship measurement.

Recent work has started to fill some of these gaps. Elmenhorst et al. [47] integrated QUIC support into OONI and reported HTTP/3 blocking through UDP disruption in Iran and IP-based QUIC blocking in China and India . Complementing this, OONI's

Authors	Year	Description of Work
Crandall et al. [19]	2007	ConceptDoppler tool for censorship detection.
Weinberg et al. [36]	2011	Proposal for a global censorship observatory.
Jones et al. [8]	2020	Challenges of measuring internet censorship at scale.
Bock et al. [6]	2021	QUIC censorship detection techniques in China.
Mishra et al. [33]	2021	Performance evaluation of HTTP/3 and censorship indicators.
GitHub/net4people [40]	2021	Issue documenting blocking of UDP 443 in Russia.
Elmenhorst [26]	2022	Study on QUIC accessibility and censorship visibility.
OONI [39]	2023	OONI report on censorship in Russia during military conflict.
OONI [41]	2023	OONI measurements showing censorship in Azerbaijan.
Reporters Without Borders [42]	2023	Report on internet censorship in Kazakhstan.
Elmenhorst et al. [49]	2025	Comparative analysis of HTTP/3 vs. HTTP/1.1 reachability.
Wendzel et al. [48]	2025	Survey highlighting gaps in encrypted protocol measurement.

Table 3.3: Literature on Limitations of Present Studies

extended 2022 dataset confirmed that QUIC blocking varies widely, with some networks allowing HTTP/3 freely while others interfere selectively, diverging from HTTPS patterns [56]. Zohaib et al [57], further demonstrated that the Great Firewall of China has evolved to inspect QUIC Initial packets and apply SNI-based filtering heuristics, marking one of the most comprehensive investigations into QUIC-targeted censorship to date . Building on cross-layer perspectives, Sengupta and Bajpai [58] analyzed interactions between QUIC, encrypted DNS, and HTTP/3, showing that censorship strategies can exploit coalesced transport behaviors, an area still largely unmeasured in many regions. Broader surveys, such as Wendzel et al. [59], emphasize that the lack of vantage points across Africa, Central Asia, and Southeast Asia remains a systemic limitation for QUIC-aware measurements.

Beyond QUIC-specific work, regional studies highlight a lack of measurement visibility in large parts of the Global South. Table 3.4 summarizes recent studies that address regional gaps in QUIC censorship and related measurement efforts.In Africa, Isah et al [60]. surveyed operators and researchers and found that resource constraints and limited infrastructure inhibit deployment of censorship measurement systems . Similarly, a recent APNIC Foundation project demonstrated that censorship in Southeast Asia, including Malaysia and Thailand, is often heterogeneous across ISPs, complicating attempts to generalize from limited vantage points [61]. Meanwhile, Pearce et al.'s [62] Monocle system showed that routing churn can alter censorship visibility, with important implications for UDP-based protocols like QUIC that are more path-sensitive . Existing platforms such as ICLab have provided global, longitudinal measurement coverage, but typically lack protocol-specific modules for QUIC/HTTP/3, leaving gaps in understanding protocol-aware blocking [63]. Finally, Ivanovic et al [64]. quantified censorship resilience at a global scale, showing that countries with fragmented or peripheral Internet connectivity may experience censorship differently than core networks, underscoring the need for diversified measurement targets .

Authors / Source	**Year**	**Focus / Contribution**
Isah et al. [61]	2020	Survey on Internet measurement in Africa; documents lack of infrastructure for censorship detection.
Niaki et al. [64]	2020	ICLab platform; global longitudinal measurement but lacking QUIC/HTTP/3-specific modules.
OONI [578]	2022	HTTP/3 censorship measurements with OONI Probe; highlights varying levels of QUIC blocking across networks.
APNIC Foundation [62]	2023	Technical report on network interference in Southeast Asia; shows heterogeneous censorship across ISPs.
Sengupta and Bajpai [59]	2024	Examines cross-layer interactions of QUIC, encrypted DNS, and HTTP/3; shows new vectors for interference.
Bhaskar et al. [63]	2024	Monocle system analysis; routing churn effects on censorship visibility, with implications for UDP/QUIC traffic.
Ivanovič et al. [65]	2024	Quantitative study on censorship resilience; shows topology impacts on regional blocking, especially in fragmented networks.
Zohaib et al. [58]	2025	Demonstrates SNI-based QUIC censorship in China's Great Firewall; shows packet inspection of QUIC Initials.
Wendzel et al. [60]	2025	Survey of Internet censorship measurement; identifies persistent gaps in regional QUIC-aware measurements.

Table 3.4: Recent Studies Addressing Regional Gaps in QUIC Censorship and Measurement

This thesis contributes toward addressing this gap by focusing on the detection of QUIC censorship in several of these non-exposed or understudied countries, thereby extending the reach of empirical censorship measurement.

3.5 Conclusion

In conclusion, the literature highlights both the advancements and the limitations in understanding QUIC censorship. While extensive studies have documented blocking behaviors in countries such as China, Iran, and Russia, many regions including Southeast Asia, Central Asia, and sub-Saharan Africa remain underrepresented due to limited vantage points and the absence of QUIC-aware measurement tools or lack of motivation. This uneven coverage restricts the ability to form a comprehensive global picture of how modern, encrypted transport protocols are being targeted.

At the same time, recent work has shown that censorship is becoming increasingly nuanced, with strategies that exploit routing dynamics, cross-layer interactions, and subtle traffic fingerprinting. These findings point to the inadequacy of legacy measurement frameworks and emphasize the urgent need for regionally diverse and protocol-specific approaches. By focusing on underexamined regions, this thesis contributes to addressing these blind spots and extends the empirical foundation necessary to evaluate the resilience of QUIC against evolving censorship practices systematically in a selected region .

Chapter 4

System Design

This chapter explains the design methodology employed to detect and analyze QUIC censorship through cross-vantage measurements. It outlines the procedural steps undertaken to identify a curated set of QUIC-enabled websites and measure their accessibility from two different geographic locations. Switzerland (as the control) and India, Malaysia, Thailand and Indonesia (as test sites). The study begins by constructing a representative dataset of QUIC-capable domains, sampled from the Tranco Top 1 Million list [48]. Each domain is tested from the Swiss vantage point using HTTP/3-enabled requests to verify support for QUIC. Domains that respond successfully over QUIC are retained to form a validated, censorship-free baseline. Following this, measurement environments are deployed one in Switzerland and the others in vantage points in target countries using virtual private servers (VPS) configured with identical network tools and system setups. These environments are tasked with probing each validated domain using both QUIC and TCP-based protocols. Key metrics such as HTTP status codes, connection outcomes, and error messages are logged. By comparing responses between the locations, the study can infer potential censorship when QUIC fails in vantage points but TCP succeeds. The chapter also details the logging schema, ethical safeguards, and decision logic used to evaluate protocol-level blocking, establishing a robust and reproducible experimental architecture for empirical censorship analysis.

4.1 Objective and Approach

This chapter presents the design approach that was adopted to investigate the presence of QUIC censorship across different geographic locations. The objective was to compare the accessibility of a selected subset of QUIC-enabled websites from four vantage points: Switzerland, where censorship was not expected, and India along with several Southeast Asian countries, which served as the primary test locations. The goal was to determine whether access over the QUIC protocol remained consistently available in both locations or was selectively disrupted.

To achieve this, a curated set of websites that supported QUIC was compiled and first validated from the Swiss environment to establish a reliable, censorship-free baseline.

These same websites were then tested from the remaining vantage points using both QUIC and traditional TCP-based protocols. By comparing results across locations and transport layers, the study aimed to identify cases where QUIC was selectively blocked while TCP remained unaffected, which indicated targeted protocol-level censorship. This chapter outlined the high-level design decisions and procedural steps that formed the foundation for the toolâs implementation in the following stages.

Switzerland was chosen as the reference environment for identifying QUIC-supported websites due to its strong commitment to internet openness and limited network-level censorship. As noted by Freedom House, the country consistently ranked high in terms of digital freedom, offering a neutral and minimally restrictive network environment for conducting protocol-specific measurements [52]. This made Switzerland an ideal vantage point for verifying the accessibility and functionality of QUIC, independent of interference from state-imposed filtering mechanisms.

Nonetheless, Switzerland enforced content-specific restrictions in certain categories, most notably online gambling. Under the Swiss Gambling Act, the federal authority Gespa maintained a blacklist of unlicensed foreign gambling sites, which were actively blocked within the country [53]. As a result, websites such as bet365, despite being technically QUIC-capable, were deliberately excluded from the baseline list to avoid legal and interpretive complications in the censorship analysis.

By using Switzerland as a control environment and carefully omitting domains subject to local content restrictions, this methodology ensured that the selected set of QUIC-enabled websites represented a reliable and censorship-neutral benchmark. This filtered baseline was then used to assess and compare accessibility from more restrictive environments.

4.2 High-Level Design Architecture

The overall design was structured in two stages. The first stage focused on finalizing a reliable set of QUIC-supported domains, which served as the input for the second stage that conducted the actual censorship measurements across different geographic locations.

Figure 4.1 depicted the domain selection and pre-screening procedure that was used to prepare targets for the measurement campaign. An initial candidate list was curated and sanitized by removing categories deemed inappropriate or out of scope (e.g., pornography and gambling). From this filtered list, probes were executed from the Swiss vantage point to assess transport support; only domains that successfully negotiated QUIC (HTTP/3) were retained for subsequent experiments. Domains that proved reachable only via TCP/TLS (HTTP/2) or that failed on both transports were excluded from further analysis to ensure that the experimental set specifically reflected QUIC-capable endpoints.

The second stage of the design involved deploying Virtual Private Servers (VPS) in selected regions such as Malaysia and India to test the accessibility of the previously verified QUIC-supported websites. For each domain, QUIC connectivity was first evaluated. If the site responded over QUIC, it was considered uncensored. If QUIC failed, a TCP/TLS probe was issued to determine whether the service remained reachable through traditional

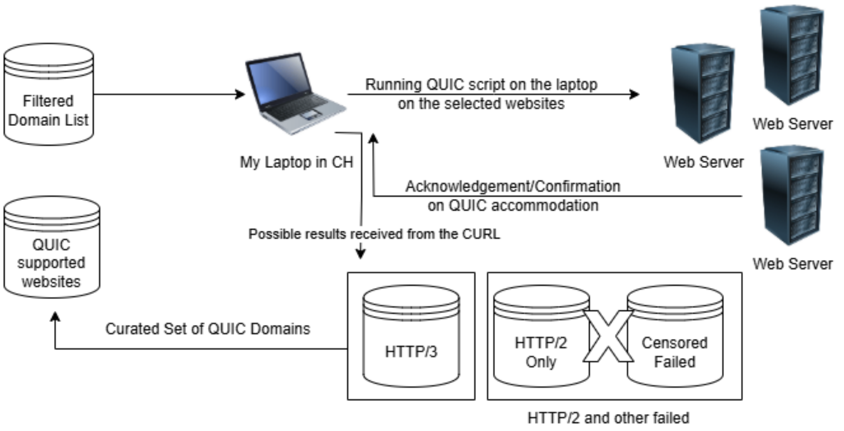

Filtered Domain List
My Laptop in CH
Running QUIC script on the laptop on the selected websites
Web Server
Web Server
Web Server
Acknowledgement/Confirmation on QUIC accommodation
Possible results received from the CURL
QUIC supported websites
Curated Set of QUIC Domains
HTTP/3
HTTP/2 Only
Censored Failed
HTTP/2 and other failed

Figure 4.1: Curated QUIC Websites for the Test

protocols. A successful TCP/TLS response combined with a failed QUIC response was treated as evidence of QUIC-specific censorship. Conversely, if both QUIC and TCP/TLS failed, the domain was considered entirely unreachable, indicating possible broader censorship not limited to QUIC.

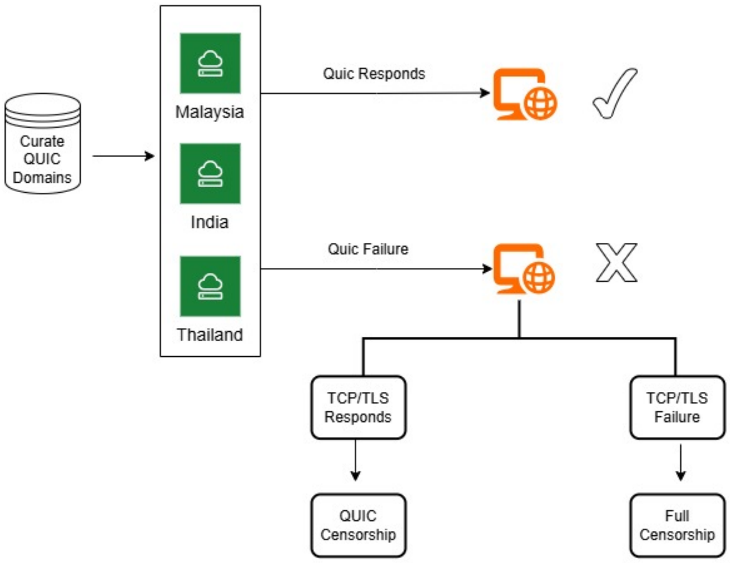


Figure 4.2: Experimental Flow on QUIC Censorship Diagnosis

Figure 4.2 illustrated the experimental decision flow that was used to diagnose QUIC-specific censorship. Domains that were pre-screened and confirmed to support QUIC from the Swiss vantage point were deployed to remote AWS EC2 vantage points in selected South and Southeast Asian locations. From each remote vantage point, the probe logic first attempted a QUIC connection: if QUIC succeeded, the site was classified as reachable via QUIC; if QUIC failed, the experiment fell back to a TCP/TLS probe to disambiguate causes. A successful TCP/TLS response following a QUIC failure was treated as a candidate for QUIC-specific interference, whereas failure of both QUIC and TCP/TLS indicated broader, likely network-level, blocking.

4.3 Website Domain Selection

To validate the measurement methodology and ensure the reliability of the VPS setup, the initial experiments were conducted using a curated list of the top 50 QUIC-enabled websites. This limited dataset allowed for controlled testing, quick verification of network behavior, and early identification of implementation issues related to protocol detection or censorship anomalies.

Once the basic functionality was confirmed and the automated measurement pipeline was established, the experiment was scaled to cover a larger subset of QUIC-enabled domains. This broader dataset enabled more comprehensive coverage and improved the robustness of censorship detection across a wider range of content categories and service providers.

This two-phase approach starting small for validation and then expanding ensured efficient development while minimizing debugging complexity in the early stages.

4.3.1 Dataset Preparation and Initial Filtering

To support the analysis of QUIC accessibility and potential censorship, an up-to-date list of the top 1 million websites was obtained from the Tranco ranking, as of May 2025 [48]. Tranco provides a reliable and reproducible ranking by aggregating multiple sources, making it suitable for research in network measurements and censorship studies.

As an initial step, the dataset requires cleaning and filtering to ensure ethical considerations and analytical relevance. Specifically, domains associated with the following categories were excluded:

- **Pornographic content**
- **Online gambling**

Removing such domains helps avoid legal, ethical, and methodological complications during measurement, while also ensuring the study focuses on general-purpose, publicly accessible web services

This filtering process helps maintain focus on general-purpose and publicly accessible websites while avoiding legal, ethical, or reputational concerns. Several publicly available domain categorization and blocklist resources were used to identify and exclude such domains. Notably:

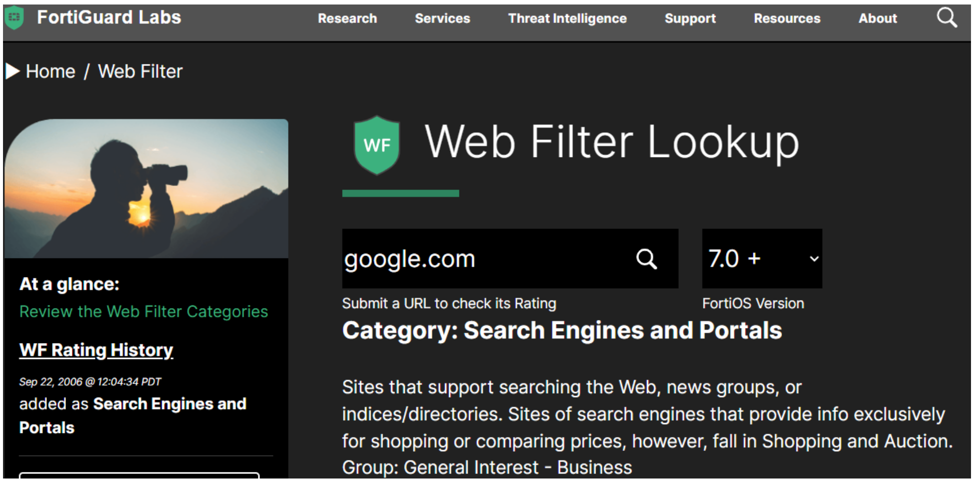
FortiGuard Web Filter enables manual checking of domain reputations and categories, such as ”Pornography”, ”Gambling”, or ”Malicious Websites”, based on Fortinet’s global threat intelligence infrastructure [49].

Webshrinker provides a domain classification API that returns structured content categories, allowing programmatic filtering of domains linked to adult, illegal, or unethical content [50].

Firebog aggregates and publishes DNS-based blocklists from a variety of trusted sources. These lists are organized by theme, such as adult content, gambling, malware, and file-sharing, and can be used to systematically filter large domain datasets in an offline or semi-automated manner [51].

These tools enabled the reliable and efficient removal of ethically problematic or irrelevant domains from the initial set, ensuring that the resulting dataset aligns with the intended scope of censorship measurements.

While the initial goal was to exclude multiple sensitive categories such as pornography, gambling, and torrenting, it was not straightforward to obtain reliable and up-to-date blocklists for categories beyond pornography and gambling. As a result, only domains associated with pornographic content and gambling were systematically identified and removed from the dataset.



Web Filter Lookup

google.com

7.0 +

Submit a URL to check its Rating

FortiOS Version

Category: Search Engines and Portals

Sites that support searching the Web, news groups, or indices/directories. Sites of search engines that provide info exclusively for shopping or comparing prices, however, fall in Shopping and Auction.
Group: General Interest - Business

At a glance:
Review the Web Filter Categories

WF Rating History

Sep 22, 2006 @ 12:04:34 PDT
added as **Search Engines and Portals**

Figure 4.3: Fortiguard URL Checker [49]

Figure 4.3 illustrates the category classification and related auxiliary information of a domain when queried through FortiGuard’s Web Filter Lookup tool. This interface provides insights into how specific domains are categorized for security and content filtering purposes, supporting the domain curation process used in this study.

As the final stage of the domain filtering process, a Python script was developed to automatically remove domains associated with pornographic content. This was achieved by comparing the initial list of domains against a publicly available blocklist sourced from

Firebog, specifically the Prigent-Adult.txt list. The script processes each domain and excludes those that appear in the blocklist, resulting in a cleaned list of pornography-free websites. This step ensures that the final dataset used for censorship measurements aligns with ethical considerations and excludes categories deemed inappropriate for the scope of this study.

To prevent false positives in our analysis, domains present on the Swiss Gambling Supervisory Authority's (Gespa) blocklist of unlicensed gambling websites were removed from the dataset of the top 1 million domains. This ensures compliance with national regulations and avoids misinterpretation of these domains as being censored for non-gambling-related reasons [53].

4.4 Filtering the QUIC Supported Websites

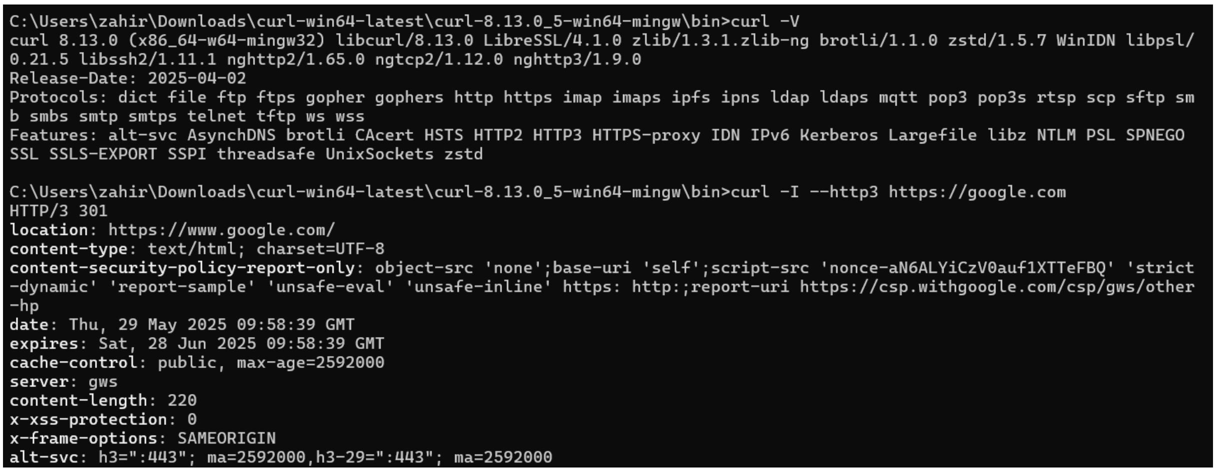
From the curated list of websitesâfiltered to exclude pornography and other sensitive categories domains were selected for further testing. To ensure that these websites genuinely supported the QUIC protocol, an initial probing phase was conducted from a control vantage point located in Switzerland, where internet censorship was not expected. As the research was based in Switzerland, this location was used to send HTTP/3 (QUIC) requests to the selected domains. Only the websites that responded successfully to QUIC probes from this control region were retained for subsequent censorship evaluation in the target countries.

To verify whether websites supported the QUIC protocol (HTTP/3), two approaches were used depending on the number of domains. For individual checks, PowerShell and the curl utility with HTTP/3 support were used to send simple requests such as `curl -I -http3 https://google.com` , which returned response headers if the site supported QUIC. For bulk verification, a script was created to automate this process by iterating over a list of domains, sending probes to each one, and recording which domains responded successfully over QUIC. This enabled efficient filtering of a larger set of candidate websites, ensuring that only domains with confirmed QUIC support were included in the final measurement set.

The following two screenshots illustrate examples of a website with QUIC support enabled and another with QUIC support disabled, respectively.

Figure 4.4 demonstrated how a QUIC-enabled domain responded to an HTTP/3 request made using the curl command. In this example, the curl client queried google.com with the HTTP/3 protocol flag (–http3), and the response confirmed that the server supported QUIC/HTTP/3 by returning standard headers such as alt-svc indicating available QUIC versions. The successful exchange of headers and the returned status code (HTTP 301) verified that the target domain accommodated HTTP/3 connections, thereby validating QUIC capability prior to its inclusion in the censorship-measurement experiments.

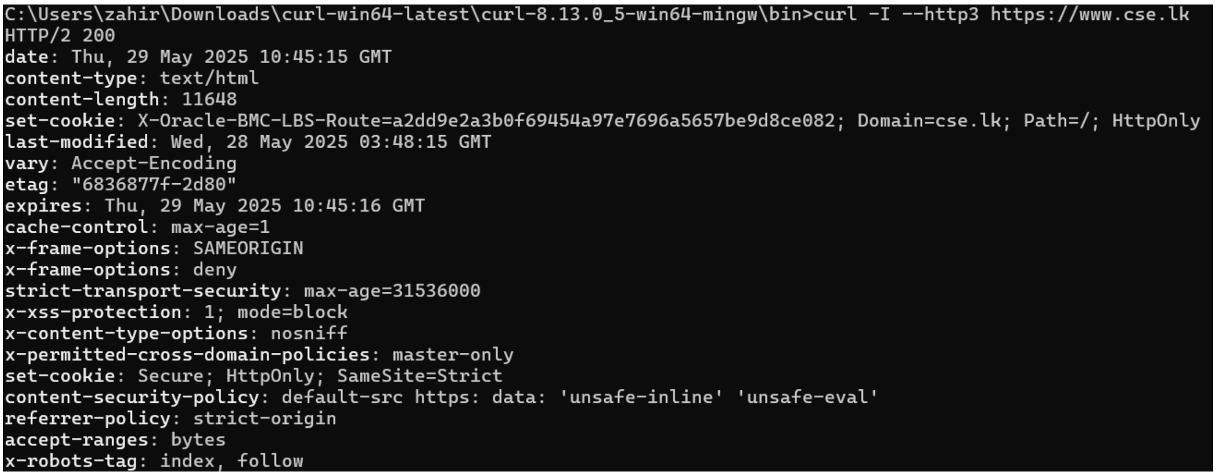
Figure 4.5 illustrated how a curl request using QUIC (–http3) behaved when the target website did not support QUIC. This was confirmed by observing that the response was



```
C:\Users\zahir\Downloads\curl-win64-latest\curl-8.13.0_5-win64-mingw\bin>curl -V
curl 8.13.0 (x86_64-w64-mingw32) libcurl/8.13.0 LibreSSL/4.1.0 zlib/1.3.1.zlib-ng brotli/1.1.0 zstd/1.5.7 WinIDN libpsl/
0.21.5 libssh2/1.11.1 nghttp2/1.65.0 ngtcp2/1.12.0 nghttp3/1.9.0
Release-Date: 2025-04-02
Protocols: dict file ftp ftps gopher gophers http https imap imaps ipfs ipns ldap ldaps mqtt pop3 pop3s rtsp scp sftp sm
b smbs smtp smtps telnet tftp ws wss
Features: alt-svc AsynchDNS brotli CAcert HSTS HTTP2 HTTP3 HTTPS-proxy IDN IPv6 Kerberos Largefile libz NTLM PSL SPNEGO
SSL SSLS-EXPORT SSPI threadsafe UnixSockets zstd

C:\Users\zahir\Downloads\curl-win64-latest\curl-8.13.0_5-win64-mingw\bin>curl -I --http3 https://google.com
HTTP/3 301
location: https://www.google.com/
content-type: text/html; charset=UTF-8
content-security-policy-report-only: object-src 'none';base-uri 'self';script-src 'nonce-aN6ALYiCzV0auf1XTTeFBQ' 'strict
-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https: http:;report-uri https://csp.withgoogle.com/csp/gws/other
-hp
date: Thu, 29 May 2025 09:58:39 GMT
expires: Sat, 28 Jun 2025 09:58:39 GMT
cache-control: public, max-age=2592000
server: gws
content-length: 220
x-xss-protection: 0
x-frame-options: SAMEORIGIN
alt-svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000
```

Figure 4.4: Success Response of QUIC-Enabled Website

served over HTTP/2 (indicated by the status line HTTP/2 200) and by the absence of an alt-svc header advertising QUIC support (e.g., h3=443).



```
C:\Users\zahir\Downloads\curl-win64-latest\curl-8.13.0_5-win64-mingw\bin>curl -I --http3 https://www.cse.lk
HTTP/2 200
date: Thu, 29 May 2025 10:45:15 GMT
content-type: text/html
content-length: 11648
set-cookie: X-Oracle-BMC-LBS-Route=a2dd9e2a3b0f69454a97e7696a5657be9d8ce082; Domain=cse.lk; Path=/; HttpOnly
last-modified: Wed, 28 May 2025 03:48:15 GMT
vary: Accept-Encoding
etag: "6836877f-2d80"
expires: Thu, 29 May 2025 10:45:16 GMT
cache-control: max-age=1
x-frame-options: SAMEORIGIN
x-frame-options: deny
strict-transport-security: max-age=31536000
x-xss-protection: 1; mode=block
x-content-type-options: nosniff
x-permitted-cross-domain-policies: master-only
set-cookie: Secure; HttpOnly; SameSite=Strict
content-security-policy: default-src https: data: 'unsafe-inline' 'unsafe-eval'
referrer-policy: strict-origin
accept-ranges: bytes
x-robots-tag: index, follow
```

Figure 4.5: Response of a Non-QUIC Website

Since manually verifying QUIC support for each website in the candidate list was time-consuming and inefficient, the process was automated using a script that programmatically issued HTTP/3 (curl –http3) header requests to each domain. The script parsed the responses to determine whether the servers replied using HTTP/3 or fell back to an earlier protocol such as HTTP/2. This automation enabled rapid and consistent classification of websites based on their support for QUIC, thereby ensuring scalability and accuracy in the selection of measurement targets.

4.5 Vantage Point Setup

To conduct the censorship measurements, India was selected as the initial vantage point, with subsequent vantage points established throughout the experiment in countries such as Malaysia, Indonesia, and Thailand. In each case, Virtual Private Servers (VPS) were rented and configured for the duration of the experiment. These servers were used to send QUIC and TCP/TLS probes to a predefined set of websites and recorded the corresponding responses. Given the lightweight nature of the measurements primarily focused on connection attempts, response headers, and basic logging, high-performance machines were not required. Thus, low-tier Linux-based VPS instances with minimal resources were sufficient.

While provisioning a VPS in India was relatively straightforward due to widespread support from cloud providers such as AWS with regional availability zones, finding a suitable VPS provider in Sri Lanka posed a greater challenge, as international cloud platforms did not typically maintain infrastructure there. Instead, local providers or smaller hosting companies were considered to establish a functional vantage point. Depending on time and resource availability, additional countries were incorporated later to enrich the analysis and increase geographic coverage.

4.6 Implementation of the Probing Tool

To carry out the censorship measurement experiment efficiently and reproducibly, a dedicated tool was developed. This tool automated the process of testing a list of QUIC-supported websites from various vantage points (e.g., VPS instances in Malaysia and India) and logged the results for later analysis. The primary objective was to detect whether QUIC-specific censorship was present by comparing the accessibility of websites using QUIC versus TCP/TLS. The tool was designed to be lightweight, efficient, and capable of producing clear logs indicating the result of each probe.

The decision to develop a custom tool stemmed from the need to:

- Systematically probe a large list of domains across multiple protocols.
- Ensure consistent request behavior across experiments.
- Log and store results in a structured format for subsequent evaluation.
- Reduce manual effort and potential human error in the measurement process.

Tool Modules

The tool designed for this experiment consists of the following core modules:

- **Input Handler**
  - Reads a file containing the final list of QUIC-supported domains to be tested.
- **QUIC Probe Module**

  - Sends an HTTP/3 (QUIC) request to each domain using a command-line utility such as `curl`.
  - Determines whether the QUIC request succeeds or fails.

- **TCP/TLS Fallback Module**
  - Initiates a fallback HTTPS request over TCP if the QUIC request fails.
  - Determines if the website is reachable over standard HTTPS to detect protocol-specific censorship.

- **Result Logger**
  - Logs the outcome of each probe, including:
    - Domain name
    - QUIC success/failure
    - TCP success/failure (if applicable)
    - Timestamp
    - Optional notes (e.g., HTTP status code or error message)
  - Stores the results in a structured format (e.g., CSV) for later analysis.

- **Rate Limiter**
  - Introduces a short delay between successive requests to avoid overwhelming remote servers and maintain ethical measurement behavior.

4.7 Measurement Workflow

The measurement workflow followed a structured logic to isolate instances of potential QUIC-specific censorship. For each curated domain under test, the automated tool was used, with Switzerland serving as the baseline vantage point that provided positive QUIC responses. Next, the same QUIC probe was sent from the remote vantage points. If this request failed, a follow-up TCP/TLS probe was issued from the same server to determine whether the domain remained reachable via traditional protocols. A case where TCP succeeded while QUIC failed was interpreted as indicative of QUIC-specific interference rather than a generic service outage or misconfiguration. This sequential probing logic ensured that observed accessibility issues were not tied to the transport protocol in use, thereby enabling a more accurate diagnosis of protocol-level censorship.

4.8 Ethical Considerations

This experiment was designed with a clear commitment to ethical research practices. No human participants or volunteers were involved, eliminating concerns related to personal data collection or informed consent. All measurements were conducted using controlled

Virtual Private Servers (VPS) deployed in selected regions, ensuring that probes originated from owned (Swiss) or rented (AWS) infrastructure. Additionally, the set of target domains was pre-filtered to exclude sensitive, illegal, or potentially harmful categories such as pornography and gambling in line with both legal and ethical standards.

The experiment did not involve probing or scanning private, uninvited, or non-consenting endpoints, nor did it attempt to exploit vulnerabilities, perform intrusive scans, or impact the functionality of the tested services. All requests were limited to standard protocol-level probes (e.g., QUIC or TCP handshake attempts) that mimicked ordinary user behavior, without attempting content retrieval or service disruption. As the study focused on protocol reachability, no payloads or personally identifiable information were exchanged, and no surveillance or logging beyond essential response metadata was performed. Furthermore, care was taken to avoid excessive frequency or volume of traffic, thereby minimizing the risk of burdening target servers. This approach ensured compliance with ethical standards for network measurement studies, including respect for user privacy, data minimization, and non-intrusiveness.

To ensure transparency and minimize potential ethical concerns, each HTTP/3 probe sent using curl included a custom HTTP header that clearly indicated the request was part of an academic research study. The header message stated the purpose of the measurement and provided a contact address for administrators wishing to opt out. This approach aimed to respect the autonomy of server operators while maintaining the integrity of the measurement process.

Chapter 5

Implementation

This chapter describes how the measurement framework designed in the previous section was put into practice. It explains the deployment of vantage points in different geographic regions, the process of selecting QUIC-enabled websites as a baseline, and the development of tools to probe these sites for HTTP/3 support. In addition, it outlines the workflow for collecting and organizing the measurement data, and discusses practical challenges encountered during large-scale probing. Together, these components show how the conceptual design was implemented to enable the analysis of QUIC availability and potential censorship across multiple countries.

This thesis employs a distributed testing setup to systematically measure potential censorship of QUIC traffic. It consists of the following components:

i. A local machine in Switzerland, which establishes a baseline of QUIC-supported websites from a non-censored environment.

ii. VPS servers deployed in Thailand, Indonesia, India and Malaysia, which perform remote probing to assess HTTP/3 availability under different regional network conditions.

iii. A comparison framework designed to analyze discrepancies between the baseline and remote measurements, thereby enabling the identification of possible censorship or protocol-specific interference across locations.

5.1 Domain Cleanup Pipeline

As a baseline, this study utilized a vantage point located in Switzerland, representing a non-censored environment, to identify domains that actively support QUIC. Starting with a dataset of one million domains obtained from the Tranco list, an established ranking of the most popular websites, an initial cleaning phase was performed to exclude domains associated with unethical or sensitive content, such as pornography and gambling sites. This was implemented as a best-effort filtering step to align with ethical research practices.

To further enhance the integrity of the baseline dataset, an additional keyword-based filtering step was applied. This process systematically removed domains containing terms associated with disallowed or ethically sensitive categories, thereby refining the list to better align with the study objectives.

5.2 QUIC Baseline Detection from Switzerland

Subsequently, the resulting domain list was divided into eleven approximately equal-sized chunks. Each chunk was processed in parallel using batch scripts that employed curl with the –http3 option to explicitly test for HTTP/3 (QUIC) support. This parallelized approach enabled the efficient probing of hundreds of thousands of domains to establish a comprehensive baseline set of QUIC-capable websites under conditions free from network interference or censorship.

To establish a reliable baseline set of domains, I have first verified QUIC support from a Swiss vantage point by checking for the presence of the HTTP/3 response header. Since HTTP/3 is specified to run exclusively over QUIC, this header serves as a practical indicator of QUIC capability. While it is possible that some servers may not advertise QUIC or may redirect to alternative endpoints, using HTTP/3 as a proxy allows us to confidently assume that the selected domains are QUIC-capable under normal conditions. This approach provides a solid and consistent baseline for measurement, even though variations in DNS resolution or server configuration across regions may influence whether the same header is observed elsewhere.

Once all the batch files had completed their execution, the resulting lists of QUIC-enabled websites from each individual chunk were consolidated into a single aggregated file. This unified dataset provided a comprehensive inventory of domains that successfully negotiated HTTP/3 connections during the baseline probing in Switzerland, serving as the foundation for subsequent comparative measurements from other vantage points.

5.2.1 Baseline Results from Swiss QUIC Probing - Initial Classification

From the initial dataset of 1,000,000 domains, a comprehensive filtering process was applied to remove obtrusive or ethically sensitive categories such as gambling and adult content which brought this down to 960,319. The remaining domains were probed from a Swiss vantage point to detect the presence of HTTP/3 headers, which serve as a reliable indicator of QUIC protocol support. This probing resulted in the identification of over 270,000 domains that explicitly advertised HTTP/3 capability. These domains form the baseline set for the subsequent analysis of QUIC censorship, ensuring that the measurement framework operates on a corpus of websites where QUIC support is verified rather than assumed.

The analysis revealed that nearly 29 percent of the examined domains exhibited QUIC support, highlighting its growing presence in the web ecosystem. This finding aligns closely with broader industry surveys, such as those by W3Techs, which report that approximately 35 percent of websites have already adopted HTTP/3. Together, these figures underscore the fact that a significant portion of prominent domains are actively transitioning toward QUIC as a next-generation transport protocol [54].

When establishing the Swiss baseline of QUIC-enabled domains, responses that did not return a direct 200 OK were still considered valid, provided they reflected normal web behavior rather than blocking. In particular, common redirect status codes such as 301 (Moved Permanently), 302 (Found), 307 (Temporary Redirect), and 308 (Permanent Redirect) were accepted, since these indicate routine server-side redirections and not censorship of QUIC traffic. Similarly, informational codes like 100 Continue or 204 No Content do not undermine the classification of a domain as QUIC-supported. As such, the baseline included domains that responded with redirects or other non-200 success codes, ensuring that the final set of QUIC-supported domains reflected genuine protocol availability rather than being filtered out by ordinary web traffic management.

5.3 CLI Tool Implementation and Execution

The tool was specifically designed to probe domains and log detailed parameters that are essential for evaluating potential QUIC censorship. For each domain, it records outcomes such as QUIC success or failure, and, in the case of failure, whether a TCP/TLS fallback succeeded. In addition, the tool captures key metadata including negotiated ALPN values, HTTP status codes, and error messages when present. These comprehensive logs ensure that both successful and failed connection attempts can be systematically analyzed. Once development was completed, the tool was deployed as a command-line interface (CLI) application on remote vantage points, allowing automated probing and collection of results for large-scale measurement experiments.

5.4 Vantage Point Deployment

To enable measurements from outside Switzerland, vantage points were deployed using Amazon EC2 instances hosted in a geographically distinct region. The instance was configured via the AWS Management Console and provisioned with a publicly accessible IP address. Secure remote access was established through SSH from my local machine in Switzerland, enabling direct control over the virtual private server (VPS). Outgoing traffic was unrestricted, allowing the VPS to initiate probes to any destination on the internet without firewall or routing constraints. This setup provided a stable and isolated environment for conducting HTTP/3 (QUIC) measurements under controlled conditions, and ensured that the results accurately reflected the network behavior observed from the VPS's geographic location.

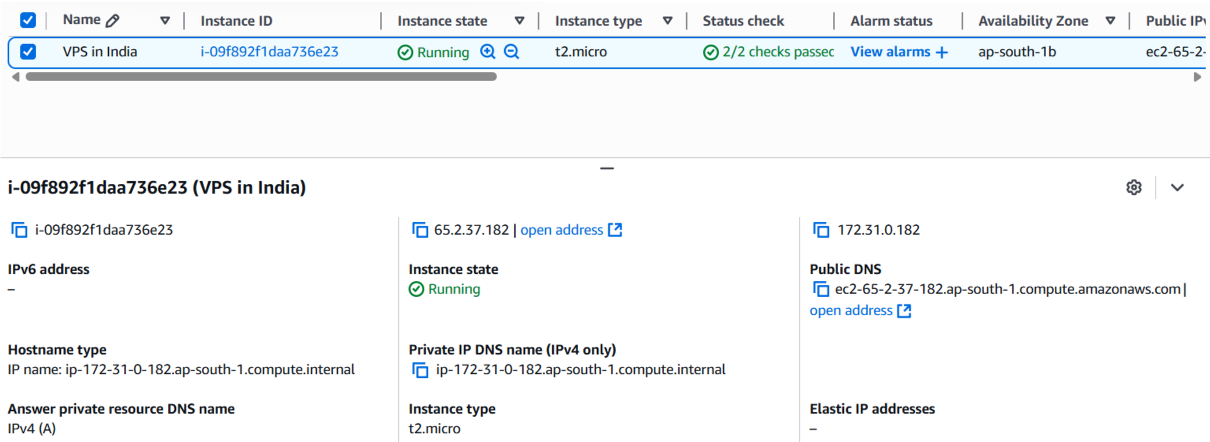
5.4.1 Vantage Point Selection

Since the core aim of this thesis is to investigate QUIC censorship in regions that have not been extensively studied, selecting geographically appropriate vantage points was a critical step. To inform this process, a regional heatmap provided by my supervisor, Thomas Grübl was used to visualize the geographic coverage of previous measurement studies. Although India had already been examined in several prior works, it was initially selected due to its proximity to Sri Lanka and the practical availability of affordable VPS infrastructure. However, to enhance the novelty and regional diversity of the dataset, additional vantage points were established in Malaysia, Thailand, and Indonesia. These countries were chosen both for their clear under representation on the heatmap and for the accessibility of reasonably priced VPS services, which made remote deployment feasible.

5.4.2 Vantage Point Service Provider Selection

For the deployment of remote vantage points, Amazon Web Services (AWS) was selected as the cloud service provider, primarily due to its extensive global infrastructure and wide geographic coverage. This distribution closely aligned with the regions identified as understudied in the supervisory heatmap, making AWS a practical choice for targeting unexplored areas. Among the available services, Amazon EC2 (Elastic Compute Cloud) instances were chosen for their flexibility, ease of configuration, and cost-effectiveness. The pricing and resource allocation offered by EC2 matched the experimental requirements, providing sufficient computational power and network access without exceeding budgetary constraints. Overall, AWS offered a reliable and scalable platform that supported both the geographic diversity and technical demands of the censorship measurement experiment.

Figure 5.1 is a snapshot of the configured EC2 instance in Amazon Web Services.



☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
☑	VPS in India	i-09f892f1daa736e23	✓ Running	t2.micro	✓ 2/2 checks passec	View alarms +	ap-south-1b	ec2-65-2-

i-09f892f1daa736e23 (VPS in India)

i-09f892f1daa736e23

65.2.37.182 | open address

172.31.0.182

IPv6 address
–

Instance state
✓ Running

Public DNS
ec2-65-2-37-182.ap-south-1.compute.amazonaws.com | open address

Hostname type
IP name: ip-172-31-0-182.ap-south-1.compute.internal

Private IP DNS name (IPv4 only)
ip-172-31-0-182.ap-south-1.compute.internal

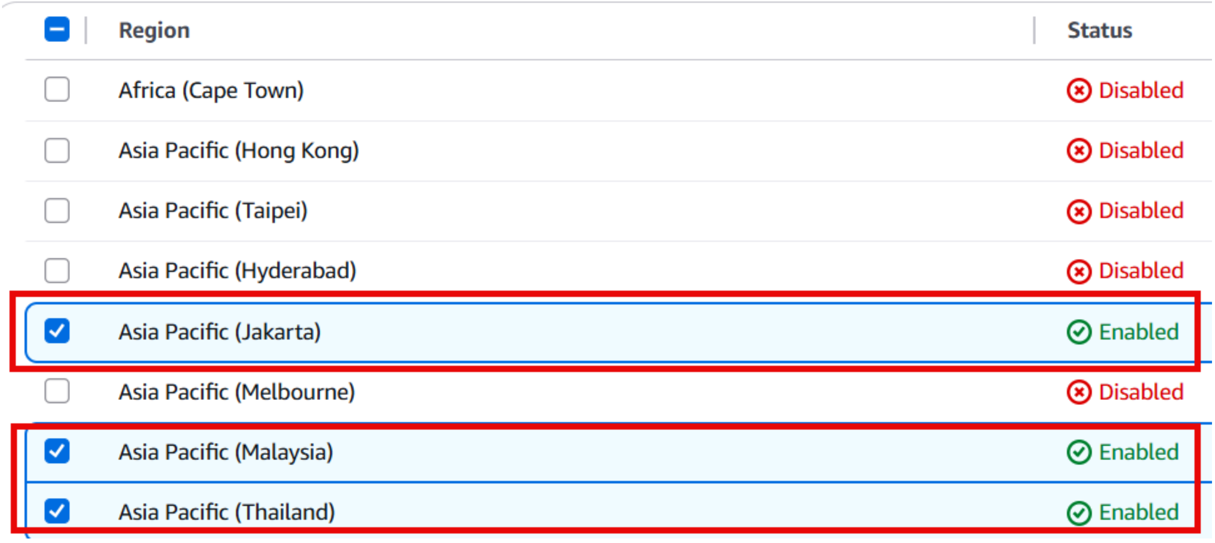
Answer private resource DNS name
IPv4 (A)

Instance type
t2.micro

Elastic IP addresses
–

Figure 5.1: EC2 Vantage Point in India

Similar to the vantage point in India, additional regions disabled by default were enabled as shown in Figure 5.2 to set up vantage points.They will follow a replicated configuration similar to initial VPS.



⊟	Region	Status
□	Africa (Cape Town)	⊗ Disabled
□	Asia Pacific (Hong Kong)	⊗ Disabled
□	Asia Pacific (Taipei)	⊗ Disabled
□	Asia Pacific (Hyderabad)	⊗ Disabled
☑	Asia Pacific (Jakarta)	✓ Enabled
□	Asia Pacific (Melbourne)	⊗ Disabled
☑	Asia Pacific (Malaysia)	✓ Enabled
☑	Asia Pacific (Thailand)	✓ Enabled

Figure 5.2: Additional Vantage Point Deployments

The vantage points in Thailand, Malaysia, and Indonesia were provisioned following the same configuration applied to the Indian vantage point. In this setup, inbound SSH access to each VPS was restricted to my control machine, while outbound traffic from the VPS is permitted without restriction to ensure unobstructed internet reachability for the probes. This arrangement provides both secure remote administration and reliable connectivity for measurements. To manage costs efficiently, the free-tier VPS instances deployed in these regions were kept in a stopped state when not in use and reactivated as needed during measurement campaigns, with configurations preserved across sessions.

Chapter 6

Experiments and Evaluation

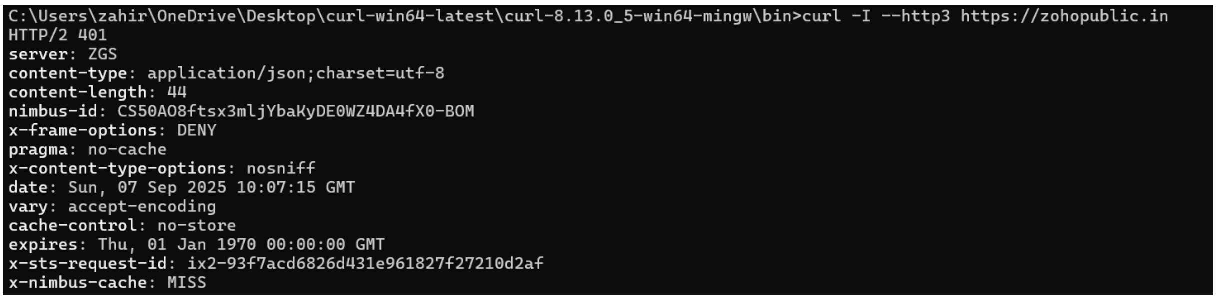
In this chapter, we present the results of our measurement campaign, which combined both baseline and vantage point testing. The Swiss vantage point was used as a baseline to verify QUIC functionality under unconstrained network conditions, while additional measurements were conducted from four VPSs located in Malaysia, Indonesia, India, and Thailand. Domains that failed or produced erroneous responses in these vantage points were subsequently re-tested in Switzerland through multiple reruns in order to filter out transient network issues and tool-related noise. This methodology enables a comparative evaluation between environments, allowing us to distinguish cases where failures are indicative of QUIC-specific interference abroad from those attributable to global reachability problems or measurement artifacts. The following sections detail the observed results per vantage point, highlight patterns of failure and fallback behavior, and provide a synthesis of findings across all measurement locations.

6.1 Validation through QUIC Sample Testing

To establish a baseline, approximately fifty domains were selected from the results of a preliminary script run. These domains had already been identified as QUIC-supported in that earlier stage, and were therefore treated as a control set. Rather than probing them anew with the main measurement tool, the previously established QUIC support information was retained. The outputs from the main tool were then compared against this control set in order to validate consistency and ensure that the probing framework produced results aligned with the earlier script-based observations.

Following are some screenshots of a failure result through aioquic and curl for http3. Aioquic is a Python library developed by the IETF community that provides a full implementation of the QUIC and HTTP/3 protocols. It allows researchers and developers to programmatically establish QUIC connections, perform handshakes, and analyze transport-layer behaviors in detail, making it particularly useful for censorship measurement and protocol testing tasks.

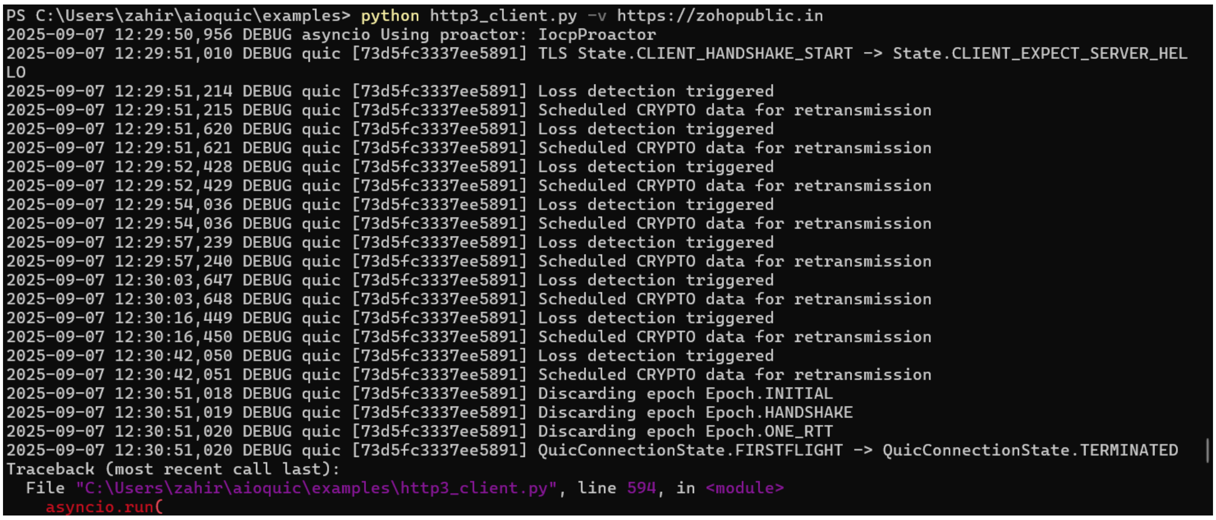
The results received through probes using aioquic isolatedly were in contrast with the tool result.Howerver the major concern is when a non quic supported domain was probed through aioquic, there is no proper output rather many errors which hinders the opportunity to evaluate what and where things went wrong.



```
C:\Users\zahir\OneDrive\Desktop\curl-win64-latest\curl-8.13.0_5-win64-mingw\bin>curl -I --http3 https://zohopublic.in
HTTP/2 401
server: ZGS
content-type: application/json;charset=utf-8
content-length: 44
nimbus-id: CS50AO8ftsx3mljYbaKyDE0WZ4DA4fX0-BOM
x-frame-options: DENY
pragma: no-cache
x-content-type-options: nosniff
date: Sun, 07 Sep 2025 10:07:15 GMT
vary: accept-encoding
cache-control: no-store
expires: Thu, 01 Jan 1970 00:00:00 GMT
x-sts-request-id: ix2-93f7acd6826d431e961827f27210d2af
x-nimbus-cache: MISS
```

Figure 6.1: HTTP/3 Curl Test on Non-QUIC Domain

Figure 6.1 shows the result of a probe executed on a domain that does not support QUIC, verified using the curl command with the HTTP/3 option. The response indicates an HTTP/2 connection with no negotiation for QUIC, confirming that the server is not QUIC-enabled. This outcome aligns with the tool's own classification, demonstrating consistency between the experimental probing framework and the independent verification using curl for HTTP/3 requests.



```
PS C:\Users\zahir\aioquic\examples> python http3_client.py -v https://zohopublic.in
2025-09-07 12:29:50,956 DEBUG asyncio Using proactor: IocpProactor
2025-09-07 12:29:51,010 DEBUG quic [73d5fc3337ee5891] TLS State.CLIENT_HANDSHAKE_START -> State.CLIENT_EXPECT_SERVER_HEL
LO
2025-09-07 12:29:51,214 DEBUG quic [73d5fc3337ee5891] Loss detection triggered
2025-09-07 12:29:51,215 DEBUG quic [73d5fc3337ee5891] Scheduled CRYPTO data for retransmission
2025-09-07 12:29:51,620 DEBUG quic [73d5fc3337ee5891] Loss detection triggered
2025-09-07 12:29:51,621 DEBUG quic [73d5fc3337ee5891] Scheduled CRYPTO data for retransmission
2025-09-07 12:29:52,428 DEBUG quic [73d5fc3337ee5891] Loss detection triggered
2025-09-07 12:29:52,429 DEBUG quic [73d5fc3337ee5891] Scheduled CRYPTO data for retransmission
2025-09-07 12:29:54,036 DEBUG quic [73d5fc3337ee5891] Loss detection triggered
2025-09-07 12:29:54,036 DEBUG quic [73d5fc3337ee5891] Scheduled CRYPTO data for retransmission
2025-09-07 12:29:57,239 DEBUG quic [73d5fc3337ee5891] Loss detection triggered
2025-09-07 12:29:57,240 DEBUG quic [73d5fc3337ee5891] Scheduled CRYPTO data for retransmission
2025-09-07 12:30:03,647 DEBUG quic [73d5fc3337ee5891] Loss detection triggered
2025-09-07 12:30:03,648 DEBUG quic [73d5fc3337ee5891] Scheduled CRYPTO data for retransmission
2025-09-07 12:30:16,449 DEBUG quic [73d5fc3337ee5891] Loss detection triggered
2025-09-07 12:30:16,450 DEBUG quic [73d5fc3337ee5891] Scheduled CRYPTO data for retransmission
2025-09-07 12:30:42,050 DEBUG quic [73d5fc3337ee5891] Loss detection triggered
2025-09-07 12:30:42,051 DEBUG quic [73d5fc3337ee5891] Scheduled CRYPTO data for retransmission
2025-09-07 12:30:51,018 DEBUG quic [73d5fc3337ee5891] Discarding epoch Epoch.INITIAL
2025-09-07 12:30:51,019 DEBUG quic [73d5fc3337ee5891] Discarding epoch Epoch.HANDSHAKE
2025-09-07 12:30:51,020 DEBUG quic [73d5fc3337ee5891] Discarding epoch Epoch.ONE_RTT
2025-09-07 12:30:51,020 DEBUG quic [73d5fc3337ee5891] QuicConnectionState.FIRSTFLIGHT -> QuicConnectionState.TERMINATED
Traceback (most recent call last):
  File "C:\Users\zahir\aioquic\examples\http3_client.py", line 594, in <module>
    asyncio.run(
```

Figure 6.2: Aioquic Verbose Log for Non-QUIC Domain

Figure 6.2 presents the verbose output generated by the aioquic client when probing a domain that does not support QUIC. The debug logs indicate repeated attempts to retransmit handshake and cryptographic data, followed by multiple loss detection events. Since the server does not respond with QUIC-compatible packets, the connection ultimately transitions from the handshake phase to termination. This behavior confirms the absence of QUIC support on the target domain and demonstrates how the tool's diagnostic output reflects failed QUIC negotiations in real time.

As an example, for the domain zohopublic.in, the results obtained from the probing tool were consistent with those from curl –http3, both confirming HTTP/3 support. However, when the domain was tested directly through aioquic without the tool's framework, the output did not provide sufficient detail, highlighting the added value of the probing tool in extracting and structuring measurement results.

domain	quic_success	quic_alpn	quic_status_code	quic_error	tcp_tls_ran	tcp_tls_success	tcp_http_version	tcp_status_code	tcp_error
google.com	TRUE	h3	301		FALSE				
facebook.com	TRUE	h3	301		FALSE				
googleapis.com	TRUE	h3	404		FALSE				
zohopublic.in	FALSE		timeout		TRUE	TRUE	HTTP/1.1	401	
zohorecruit.com	FALSE		timeout		TRUE	TRUE	HTTP/1.1	301	
zohosites.com	FALSE		timeout		TRUE	TRUE	HTTP/1.1	301	

Table 6.1: QUIC and TCP Connectivity Results for Representative Domains.

Table 6.1 presents sample log entries generated by the measurement tool for a set of representative domains. Each probe record includes detailed fields such as QUIC success status, negotiated ALPN version, HTTP status code, and any connection errors. For domains where QUIC probing failed, the tool automatically performed a fallback TCP/TLS test and logged the corresponding results, including protocol version, HTTP response code, and error type. This structured logging approach enabled systematic comparison between QUIC and TCP connectivity to identify potential cases of QUIC-specific interference.

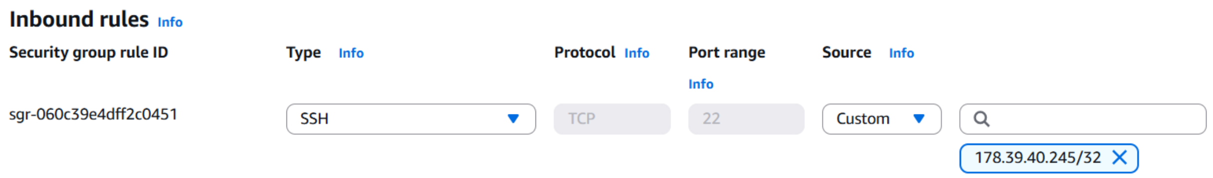
6.2 Baseline Validation with the Custom QUIC Measurement Tool

In the initial stage of this study, a lightweight script was employed to perform a base line screening from Switzerland by sending HTTPS requests with curl and identifying the presence of HTTP/3 (h3) header elements. This provided a preliminary indication of which domains appeared to support QUIC. While this approach was useful for quickly narrowing down candidates, it had clear limitations, as curl implements its own fallback mechanisms and does not always expose handshake failures directly. To achieve more accurate and fine-grained results, the subsequent screening was conducted using the custom probing tool developed in this thesis, which is built on the aioquic library. This tool allows direct inspection of the QUIC handshake, logging both successes and detailed failure reasons, and thereby offers a more reliable basis for identifying QUIC-specific blocking and optimizing the measurement process.

Running the custom tool from the Swiss baseline further refined the dataset by filtering out domains that consistently produced timeouts or errors. As a result, the initial set of 270,870 domains was reduced to 256,951 valid QUIC test domains, effectively eliminating 13,920 problematic entries. This purification step ensured that only stable and verifiable domains were forwarded to the VPS vantage points for subsequent measurements.

6.3 Probing on Vantage Points

For security hardening, inbound access to each VPS was restricted through security group rules that allowed only SSH connections from Swiss workstation IP address. This ensured that no other external sources could directly access the servers. Outbound traffic was left unrestricted, as it was required for probing domains across the internet and thus necessary for the measurement tasks.

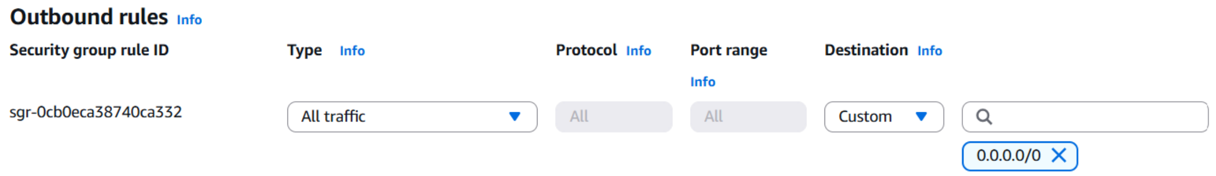


Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	
sgr-060c39e4dff2c0451	SSH	TCP	22	Custom	178.39.40.245/32

Figure 6.3: Inbound Rule Configuration on VPS Security Groups

Figure 6.3 illustrates the inbound rule configuration applied to the AWS EC2 security group used for the VPS instances. The rule permits SSH traffic over port 22 using the TCP protocol, restricted to my laptop specific IP address for secure access. This configuration allowed authenticated connections from the local computer to the remote EC2 instance, ensuring controlled administrative access while maintaining network security and minimizing unauthorized entry risks.



Outbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Destination Info	
sgr-0cb0eca38740ca332	All traffic	All	All	Custom	0.0.0.0/0

Figure 6.4: Outbound Rule Configuration on VPS Security Groups

Figure 6.4 shows the outbound rule configuration of the AWS EC2 security group associated with the VPS instances. The rule permits all outbound traffic on any port and protocol to any destination IP address (0.0.0.0/0). This configuration enables the VPS to initiate connections to external servers and services across the internet, which is essential for performing probing activities and collecting measurement data from various web domains.

Following the baseline filtering in Switzerland, the next stage involved deploying the measurement tool to remote vantage points. Four VPS instances were provisioned in India, Thailand, Malaysia, and Indonesia to capture regional variations in QUIC availability and censorship. The cleaned domain list obtained from the Swiss baseline was securely transferred to each VPS, and the probing tool, along with aioquic and all required dependencies, was installed and verified prior to execution. This ensured a consistent environment across all vantage points and enabled systematic probing of the refined dataset under diverse network conditions.

6.4 Challenges Encountered During Probing

During the large-scale measurement experiments, several technical and operational challenges were encountered that affected the stability and performance of the probing process. These challenges arose primarily due to the varying computational capacities of the VPS instances, the size of the domain datasets, and the network conditions across different vantage points. This section outlines the key issues observed during execution such as probe freezes, incomplete runs, instance limitations, and time-out constraints and describes the mitigation strategies implemented to ensure the continuity, reliability, and integrity of the collected results.

6.4.1 Probe Freeze Amidst a Large Input of Domains - Full List

One of the main challenges encountered during probing was the instability of the VPS instances when handling very large input files. In particular, when the cleaned domain list of over 250,000 entries was provided as a single file, the probe frequently froze midway through execution. The process often stopped at varying percentages of completion, which not only produced incomplete measurements but also resulted in a significant loss of time, as runs had to be restarted repeatedly. This issue was attributed to the resource strain imposed on the VPS when managing such a large dataset in a single run, which highlighted the need for a more segmented approach to ensure reliable execution.

Interestingly, this problem did not occur when running the tool from the Swiss vantage point, where the experiments were conducted on my laptop. Even with the full dataset of domains and a relatively high level of concurrency, the probing process executed smoothly without interruptions. The local environment was able to handle the workload efficiently, and the measurements completed in a timely manner, highlighting the performance gap between my setup and the constrained resources available on VPS instances.

6.4.2 Probe Stalls Before Completion

To avoid probing the full domain list at once, the dataset was divided into smaller chunks of 20,000 domains, which were executed sequentially one after the other. This approach significantly improved stability, and most of the chunks completed successfully. However, a few runs became stranded on the final domain and failed to terminate properly. Since the same tool and configuration executed flawlessly on other chunks, this behavior cannot be attributed to a tool-related issue but rather to external factors in specific runs.

As a solution to this issue, the tool was modified to enable on-the-go logging, ensuring that results were written continuously during the probing process rather than only at completion. This adjustment meant that even if a run became stranded on the final domain, all progress up to that point was safely recorded. Implementing this change eliminated the risk of losing entire batches of results and proved effective in maintaining the integrity of the collected data.

6.4.3 AWS Instance Constraints

Another challenge encountered was the limited capacity of the t2.micro VPS instances when running the probe at higher concurrency levels. Although the tool itself was optimized for efficiency, the restricted vCPU and memory resources of the t2.micro type were insufficient for sustaining continuous high-volume probing. AWS t2 instances operate on a CPU credit system, where credits are accumulated during idle periods and consumed when the CPU is under load. Once the available credits are exhausted, the instance performance is throttled, which can significantly slow down or destabilize long-running tasks such as large-scale probing. This limitation made the t2.micro less suitable for extended high-concurrency experiments, highlighting the trade-off between cost-efficiency and performance when selecting cloud infrastructure for network measurements as a solution better AWS solutions such as t3.micro or higher could be used though it would come with a cost which was not quite necessary for this experiment as there were work arounds.

6.4.4 Time-out Factor During Probing

To ensure that the probing process did not hang indefinitely on unresponsive domains, timeout mechanisms were introduced in the tool. Specifically, a maximum wait time of 12 seconds was configured for QUIC connections and 5 seconds for TCP/TLS connections. These limits allowed the tool to skip problematic domains after a reasonable delay, thereby maintaining progress and ensuring that large-scale measurements could complete efficiently.

6.5 Results Retrieval and Consolidation

After completing the probing runs across the VPS vantage points, the results from the 13 domain chunks were collected and securely transferred back to my Swiss VPS for centralized analysis. Consolidating the outputs in a single environment made it possible to evaluate the preliminary results in a consistent and controlled setting. This step ensured that all measurements could be compared side by side, allowing for a first assessment of regional differences in QUIC reachability and the effectiveness of the probing methodology.

6.6 Research Ethics in Experimental Design

All measurements in this study were conducted with careful attention to ethical considerations. Each probe request included a custom header message explicitly stating the research purpose and providing an option for website operators to opt out if desired. A dedicated email address, zahir.wazeer@proton.me , was created specifically for this purpose; however, no opt-out requests were received during the measurement period. To minimize server load and ensure responsible testing, probes were executed with deliberate gaps between consecutive attempts rather than being sent in bulk.

The concurrency level of the measurements was intentionally limited to fewer than twenty parallel probes, even though higher probing speeds were technically feasible. This limitation was applied to prevent overloading the AWS EC2 instances, which operated under constrained computational capacity, and to further reduce potential strain on target servers. Additionally, strict timeout limits were enforced to avoid prolonged connections, ensuring that each probe terminated within a reasonable timeframe. All measurements were carried out solely by the researcher using AWS EC2 vantage points, without the involvement of any third parties, thereby maintaining ethical integrity and minimizing the overall network impact.

6.7 Initial Probe Run and Results

From an initial set of approximately one million domains, a lightweight screening script identified 270,871 domains as potentially QUIC-supported. This set was subsequently refined using the custom probing tool developed in this thesis, which filtered out 13,920 unstable or erroneous cases. The resulting 256,951 domains formed the final, high-confidence baseline list that was used for cross-country probing from the VPS vantage points.

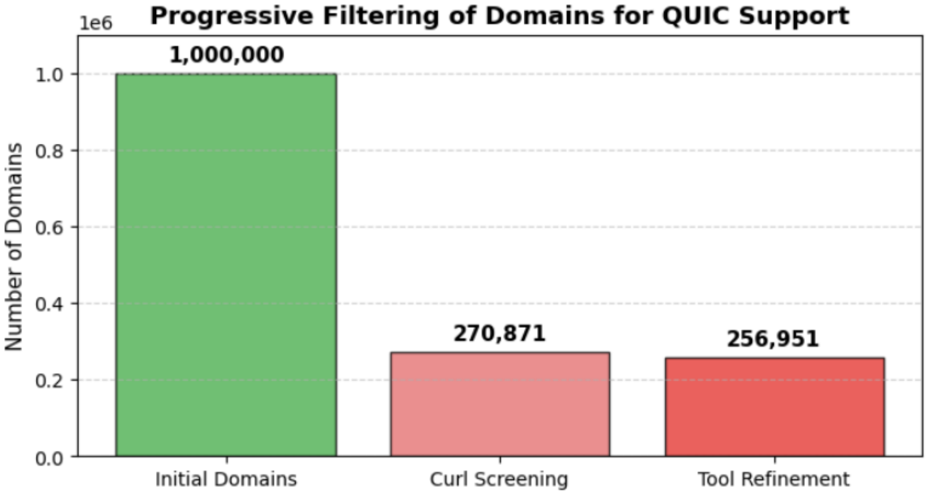


Figure 6.5: Quic Probe List Screening

Figure 6.5 compares the progression of domains across different validation stages. The first column shows the initial number of domains, the second indicates those detected as QUIC-supported using the light script, and the final column presents the QUIC-supported domains verified through the full probing tool.

The refined list of 256,951 QUIC-supported domains obtained through the tool-based screening was subsequently probed from all four VPS vantage points in India, Malaysia, Thailand, and Indonesia. Each VPS run produced a record of both successful QUIC negotiations and failures, enabling the extraction of domain-level outcomes for further analysis. These results provided the basis for identifying inconsistencies across regions and isolating cases where QUIC failures may indicate censorship.

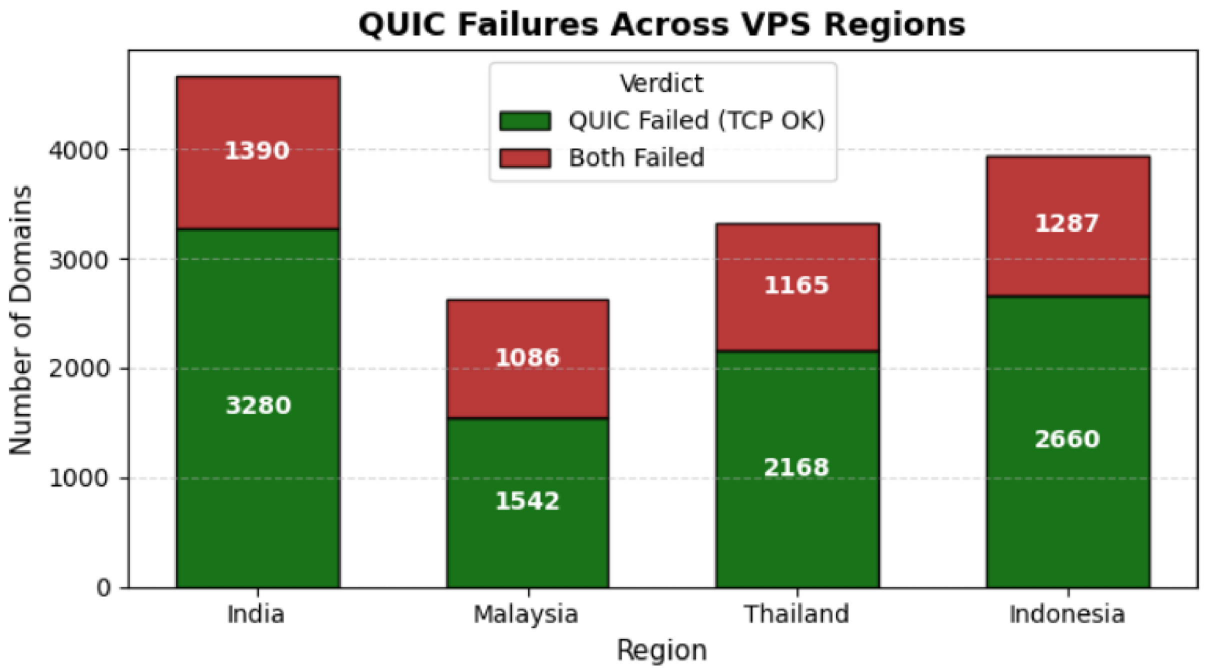


Figure 6.6: QUIC Probe on Vantage Points

Figure 6.6 illustrates the distribution of QUIC failures across different VPS regions. The green portions represent domains where QUIC failed but TCP succeeded, indicating QUIC-specific interference, while the red portions correspond to domains where both QUIC and TCP failed, suggesting complete network or site-level blocking. This comparison highlights regional variations in the extent and nature of QUIC-related censorship.

6.8 Secondary Screening of Extracted Results from VPS's

To improve the reliability of the measurements and reduce the influence of transient errors, all domains that failed, timed out, or produced erroneous results on each VPS were systematically re-tested from the Swiss baseline. Specifically, these domains were rerun ten consecutive times on the Swiss laptop, which had previously confirmed support for QUIC. The purpose of these repeated executions was to differentiate between genuine cases of QUIC disruption abroad and failures arising from temporary anomalies, server-side instability, or limitations of the probing tool. By aggregating the outcomes of these reruns, a finalized set of results was derived, enabling a more rigorous classification of domains into consistently accessible, consistently unreachable, or unstable categories. This step serves as an important validation layer, ensuring that subsequent interpretations of QUIC censorship are based on more robust evidence.

The heatmap in Figure 6.7 compares the distribution of domains that consistently succeeded in QUIC during Swiss reruns against those that exhibited mixed outcomes. India stands out with the largest number of domains in the QUIC Success category, strongly indicating that failures observed from the Indian vantage point were not due to global

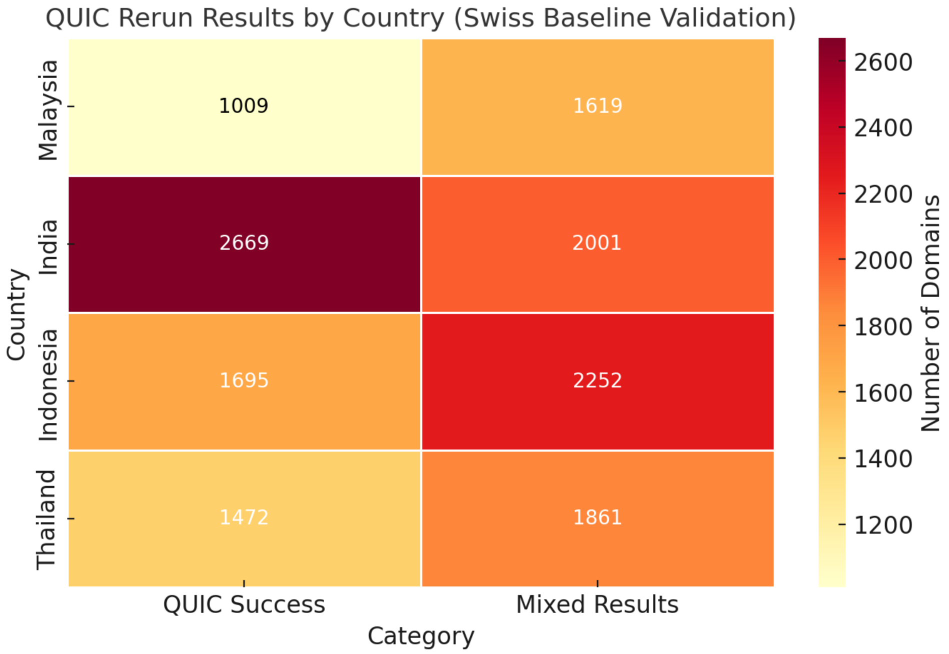


Figure 6.7: Swiss Baseline Validation of VPS Results

reachability issues but rather suggest targeted interference with QUIC traffic. In contrast, Indonesia and Thailand show higher proportions of Mixed Results, implying that a substantial share of failures may be attributable to transient network conditions, server-side instability, or tool limitations, although the presence of consistently accessible domains still points to possible censorship effects. Malaysia shows the lowest count of QUIC Success domains relative to Mixed Results, suggesting that QUIC-specific interference was less systematic there, with many failures likely stemming from non-censorship factors. Overall, these results highlight regional variation in how transport protocols such as QUIC are treated, with India presenting the strongest evidence of protocol-specific disruption.

Figure 6.8 compares the baseline probing of 256,951 QUIC-supported domains across all vantage points with the Swiss rerun outcomes of the initially failed subset. The grey bars indicate the full probing space, while the colored stacked bars highlight the revalidation results in Switzerland: domains that consistently succeeded across ten reruns (green: "Swiss Success") and those with inconsistent outcomes (red: "Swiss Mixed"). The results show that these discrepancies represent less than 2% of the total probing space, underscoring the robustness of the baseline results. Importantly, a large share of these discrepancies were confirmed as stable QUIC successes in Switzerland, strongly suggesting that their original failures in the VPS vantage points were caused by censorship or region-specific interference rather than protocol instability. In contrast, the "Swiss Mixed" domains, which displayed erratic behavior, were discarded as they are more likely attributable to transient errors or noise. Finally, the "Swiss Success" set will be rerun in the VPS vantage points to remove any remaining false positives and establish a final, censorship-confirmed set of results.

Figure 6.9 presents the classification breakdown of the re-probed domains across the four vantage points (India, Malaysia, Thailand, and Indonesia) after ten repeated runs on the

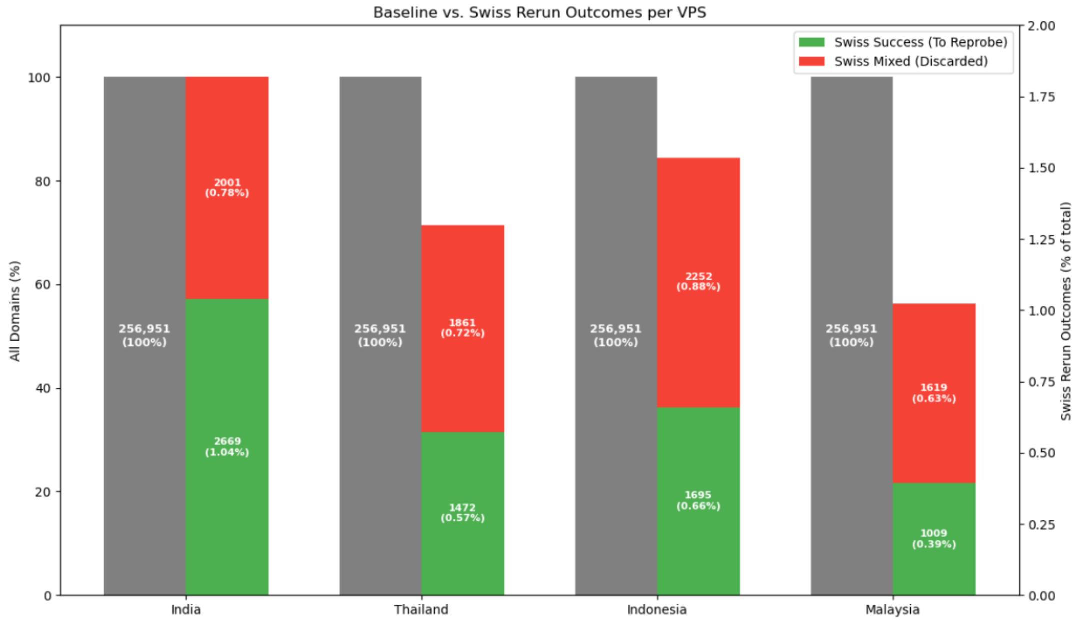


Figure 6.8: Validation on Swiss Vantage Point on VPS Failures

Swiss baseline. The stacked bar chart reflects both the absolute number of domains and their corresponding percentage distribution across four categories

A clear observation is that QUIC Success dominates in all vantage points, representing the majority of domains in each region. For example, India shows 2235 domains (83.7%) as QUIC-accessible, while Malaysia (648 domains, 64.2%), Thailand (942 domains, 64.0%), and Indonesia (994 domains, 58.6%) also demonstrate significant shares of successful QUIC connectivity. These results validate that, despite earlier failures observed at the vantage points, a large portion of these failures were false positives introduced by transient network conditions or probing artifacts.

The Both_fail category, representing domains that failed consistently at both QUIC and TCP levels, captures cases that are most plausibly attributable to network-layer blocking or site unavailability. India recorded 260 domains (9.7%), Malaysia 98 domains (9.7%), Thailand 166 domains (11.3%), and Indonesia 282 domains (16.6%). The higher percentages in Thailand and Indonesia point to stronger network-level filtering or infrastructural reachability issues in those regions compared to India and Malaysia.

The TCP_OK category highlights domains that failed under QUIC but were consistently accessible under TCP/TLS, representing the strongest candidates for QUIC-specific censorship. Although numerically smaller, these cases are crucial for establishing protocol-aware interference. India reported 163 such domains (6.1%), Malaysia 107 (10.6%), Thailand 133 (9.0%), and Indonesia 106 (6.3%). This provides strong evidence that, in addition to generic failures, targeted discrimination against QUIC traffic persists in multiple vantage points, aligning with prior findings of UDP- and protocol-specific filtering in other regions.

Finally, the Mixed Results category captures domains that alternated between QUIC failures and successes across the ten runs. India had 11 (0.5%), Malaysia 156 (15.5%), Thai-

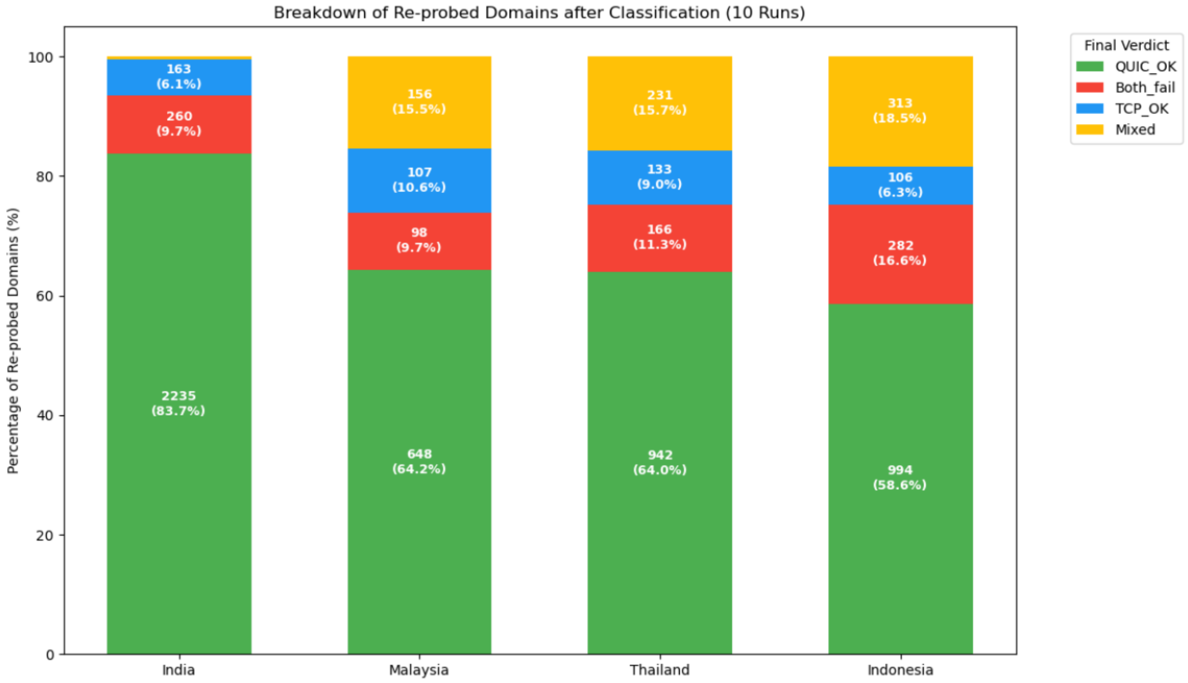


Figure 6.9: Final Classification After 10x Runs on VPS's

land 231 (15.7%), and Indonesia 313 (18.5%). Given their instability and inconsistency, these cases have been classified as false positives and are excluded from further analysis. Their presence nonetheless highlights the measurement noise and transient behaviors inherent in large-scale probing, particularly in environments with dynamic routing, load balancing, or ISP-specific interference.

Figure 6.10 visualizes the relative magnitude of QUIC-specific censorship (TCP_OK) observed across the four vantage points India, Malaysia, Thailand, and Indonesia as a proportion of the total 256 951 QUIC-enabled domains probed during the experiment. Each bubble's area corresponds to the percentage of domains in which QUIC failed but TCP/TLS succeeded, signifying potential cases of protocol-level discrimination against QUIC. This figure thus captures the extent and comparative severity of QUIC-specific interference across regions while normalizing the counts to the full experimental corpus.

The results indicate that India exhibited the highest share of TCP_OK cases, with approximately 163 domains representing 0.063% of the total test space showing clear evidence of QUIC blocking while remaining reachable over TCP. Thailand follows closely, recording 133 domains (0.052%) in this category, while Malaysia and Indonesia display slightly smaller proportions, 107 (0.042%) and 106 (0.041%) respectively. Although these values constitute less than one-tenth of a percent of the global baseline, their persistence across multiple vantage points and measurement rounds strongly suggests targeted filtering of QUIC packets rather than random transport anomalies. The uniform presence of TCP_OK cases across all four regions reinforces the interpretation that such interference is geographically distributed rather than isolated to a single network operator.

These results underline that QUIC-specific censorship, while quantitatively limited, remains non-negligible in scale and strategically focused. The fact that these TCP_OK

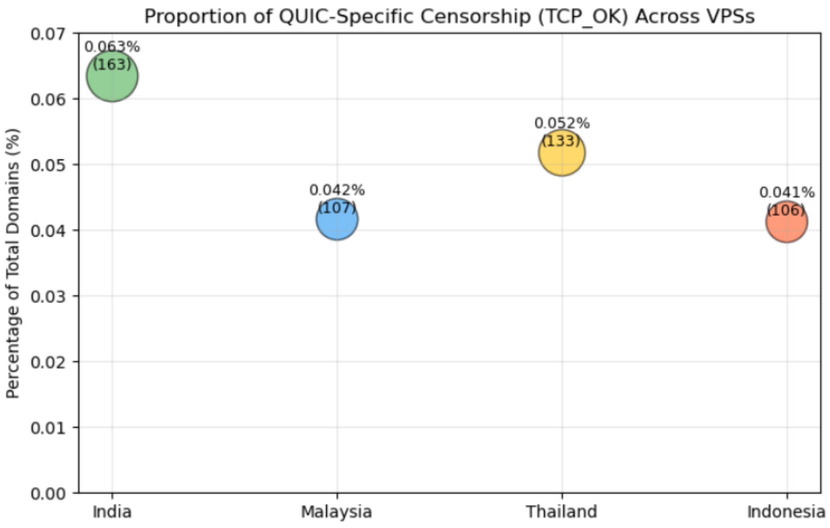


Figure 6.10: Validation on Swiss Vantage Point on VPS Failures

instances survived both the Swiss validation and multiple reruns strengthens their credibility as genuine censorship events rather than transient connectivity failures. Moreover, when considered against the total population of over a quarter million verified QUIC-supported domains, the small but recurring percentages highlight the subtle nature of modern censorshipâtargeted enough to evade aggregate detection, yet consistently reproducible under controlled measurement. Consequently, Figure 6.10 serves as an empirical confirmation that selective suppression of QUIC traffic persists in the studied regions, warranting further investigation into the filtering mechanisms and middlebox behaviors responsible for these protocol-level anomalies.

6.9 Final Validation through Packet Capture Information

To strengthen the integrity of the validation process, a post-probe verification step was conducted for all domains that initially failed QUIC connections. Each failed domain was programmatically resolved using a Python-based DNS resolution script, generating a list of corresponding IP addresses that could be directly correlated with the captured packets. This facilitated efficient filtering and matching of the Wireshark (tshark) packet captures to the respective QUIC-failed domains, enabling precise identification of handshake behaviors and termination points. By aligning the DNS-resolved IPs with the observed network traces, the process ensured that failures were accurately classified as genuine censorship events rather than false positives arising from DNS changes, transient connectivity issues, or unrelated server behavior.

Following the validation process using Wireshark handshake analysis and DNS correlation, several instances initially classified as QUIC failures were identified as false positives (FPs) that is, domains where QUIC connections ultimately succeeded despite the automated probe indicating failure. The degree of such misclassification varied across vantage points, reflecting regional network characteristics and transient measurement conditions.

For Malaysia, out of 107 domains initially labeled as QUIC-failed, 3 were later confirmed to be QUIC-successful, yielding a false positive rate (FPR) of approximately 2.8%, with 97.2% (104 domains) verified as genuine QUIC failures indicative of censorship or consistent interference. In the case of India, 13 of 163 domains (8.0% FPR) exhibited successful QUIC handshakes upon revalidation, while the remaining 150 domains (92.0%) maintained failure patterns consistent with QUIC-specific blocking. For Indonesia, 8 out of 106 domains (7.5% FPR) were false positives, leaving 98 domains (92.5%) confirmed as authentic QUIC failures. Finally, Thailand recorded 6 out of 133 domains (4.5% FPR) as false positives, whereas 127 domains (95.5%) were reaffirmed as QUIC-failed cases through packet-level inspection.

Overall, the validation indicates that while a small portion of domains initially marked as QUIC failures were later found to be transient or misclassified due to network or server-side conditions, the overwhelming majority (over 92%) of the results across all vantage points represent true instances of QUIC-specific censorship. These findings reinforce the robustness of the automated probing methodology and demonstrate the critical role of handshake-level verification in filtering out spurious measurements and achieving higher accuracy in censorship detection studies.

Country	Domains Probed	QUIC Success (FP)	FP Rate (%)
Malaysia	107	3	2.8
India	163	13	8.0
Indonesia	106	8	7.5
Thailand	133	6	4.5

Table 6.2: False Positive (FP) Rates During QUIC Revalidation

Table 6.2 presents the false positive (FP) rates observed during the QUIC revalidation phase across all vantage points. The table summarizes the number of domains initially marked as QUIC-failed but later confirmed as QUIC-successful, illustrating that false positives remained low in each regionâranging from 2.8% in Malaysia to 8.0% in India thereby confirming the overall reliability of the initial censorship classification.

6.10 Port Fuzzing

Following the validation phase, the domain sets were refined by removing those identified as false positives i.e., domains that successfully completed QUIC handshakes despite being previously classified as failed. The remaining subset thus represented domains consistently exhibiting QUIC failure patterns across multiple revalidation attempts, indicating a stronger likelihood of true censorship or transport-level interference. To further investigate whether the observed failures were port-specific or protocol-specific, a follow-up experiment was conducted using an alternative QUIC port.

In this extended test, the revalidated domain list was re-probed using port 8443, a commonly used alternative QUIC and HTTPS port. This approach aimed to determine whether the detected blocking behavior was strictly limited to the default QUIC port (443/UDP) or generalized to other QUIC-capable ports as well. By redirecting traffic through port 8443, the experiment sought to uncover any port-based filtering mechanisms or policy-driven restrictions that might be selectively applied by middleboxes or firewalls.

To capture and analyze the resulting traffic patterns, packet capture (PCAP) sessions were initiated in parallel with the probing process. The captured traces provide a granular view of handshake attempts, packet exchanges, and connection termination behaviors, offering valuable insights into whether failures persisted due to generic UDP or QUIC-level filtering, or if they were influenced by port-specific firewall rules. The packet-level evidence collected from these captures forms the basis for the subsequent analysis phase, where differences in handshake progression, packet loss, and connection resets will be interpreted to distinguish between censorship-induced interference and network-level anomalies. To determine whether QUIC blocking in India was limited to port 443 or protocol-specific, an additional probing experiment was performed on UDP port 8443 using the same 150 domains that had previously failed QUIC handshakes on port 443. The probes were executed with a concurrency level of 1 to ensure high precision and avoid overloading the vantage point or remote servers. The resulting packet capture contained 2,583 packets corresponding to 82 distinct probed domains that were successfully transmitted from the client (172.31.0.182). Although all 150 domains were targeted, some were not present in the capture due to DNS resolution issues or capture timing limitations. For the observed traffic, every attempt included outgoing QUIC Initial packets from the client, yet no corresponding server Initial, Retry, or Handshake responses were recorded. Moreover, no ICMP Port Unreachable or forged reset packets were detected, indicating that the packets were silently dropped rather than explicitly rejected. Overall, none of the 82 captured connections (0%) completed a QUIC handshake, while 100% exhibited no server response. This consistent absence of replies across all visible attempts confirms that QUIC traffic remains systematically filtered at the network layer, regardless of the destination port. The failure to elicit any QUIC handshake on 8443 despite successful TCP connectivity on 443 provides strong evidence that the interference mechanism targets the QUIC protocol itself, not a specific port number.

To examine whether QUIC interference observed on UDP/443 extended to alternate ports, an additional validation was performed on UDP/8443 across all four regional vantage pointsâIndia, Malaysia, Indonesia, and Thailand. Each vantage probed the same set of domains previously classified as QUIC-failed on port 443, using a controlled concurrency level of one to ensure precise packet timing and minimize load on both the vantage and destination servers. The resulting packet captures yielded consistent patterns across all regions. In India, 150 domains were probed, of which 82 appeared in the capture, while Malaysia (104 probed, 66 analyzed), Indonesia (98 probed, 65 analyzed), and Thailand (127 probed, 76 analyzed) exhibited comparable participation ratios. In every vantage point, QUIC Initial packets were successfully transmitted from the client, confirming that outbound UDP traffic was not locally restricted. However, no corresponding server Initial, Retry, or Handshake responses were recorded in any of the four captures, and no ICMP Port Unreachable or injected reset packets were detected.

The uniform absence of QUIC handshakes across all vantage points provides compelling evidence that the blocking mechanism is protocol-based rather than port-specific. Even when the QUIC transport was shifted from the standard port 443 to 8443, the traffic was silently dropped without any visible server acknowledgements, suggesting a consistent filtering policy applied to all QUIC-encoded UDP flows. This cross-regional consistency indicates that the interference likely occurs at the network or ISP level, where middleboxes identify and suppress QUIC packets based on recognizable handshake patterns rather than port numbers. Collectively, these findings confirm that in India, Malaysia, Indonesia, and Thailand, QUIC remains systematically censored or deprioritized independent of port configuration, reinforcing the inference that the observed blocking constitutes a form of protocol-level interference rather than a localized or server-side configuration issue.

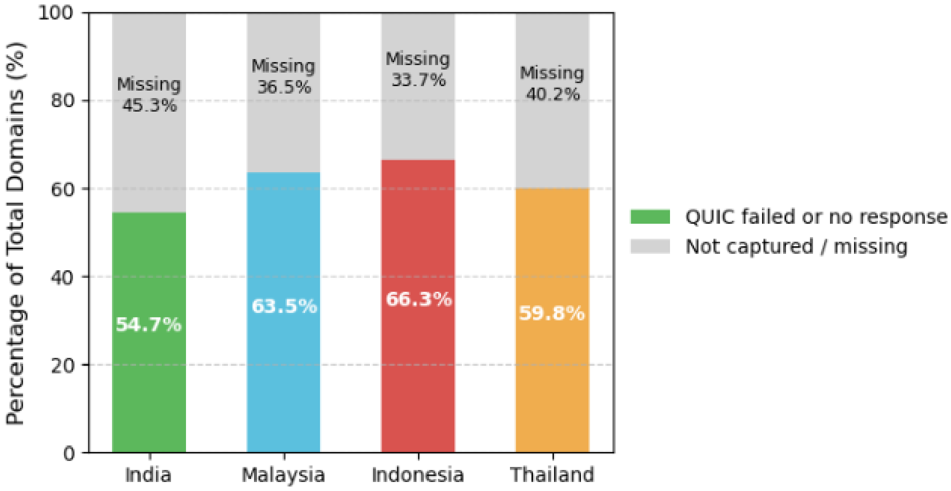


Figure 6.11: QUIC 8443 Probing Results Across Regional Vantage Points

Figure 6.11 illustrates the outcome of QUIC validation experiments conducted on UDP port 8443 across the four regional vantage points. Each colored bar represents the proportion of domains that failed to complete a QUIC handshake, while the gray segment denotes domains not captured in the packet trace due to DNS resolution or timing limitations. The figure clearly shows that all vantage points exhibited complete QUIC failure with no successful handshakes, confirming that QUIC traffic remains uniformly blocked across regions, independent of port configuration.

Chapter 7

Summary, Conclusions, and Future Work

7.1 Summary of Experiments and Evaluation

This study implemented a multi-stage experimental evaluation of censorship, to measure and validate QUIC-specific censorship across diverse regional networks. Starting from an initial dataset of over one million domains, a Swiss baseline screening reduced the corpus to 256,951 stable and QUIC-supported targets through systematic filtering and verification using a custom-built aioquic-based measurement tool. The refined list was subsequently deployed across four AWS EC2 vantage points located in India, Malaysia, Indonesia, and Thailand, each configured under identical conditions with strict security group controls and outbound access for unrestricted probing. To mitigate transient anomalies and ensure result integrity, all initial failures from these vantage points were revalidated ten times on the Swiss baseline, isolating consistent QUIC failures from ephemeral network noise.

Through this process, domains were classified into four diagnostic categories QUIC OK, TCP OK (indicative of QUIC-specific interference), Both Fail (network-layer blocking), and Mixed with the TCP OK group representing the strongest evidence of protocol-aware filtering. Subsequent packetlevel analysis using Wireshark confirmed that over 92% of QUIC failures were genuine, as client Initials were transmitted without receiving corresponding server responses, ruling out random instability. The follow up port fuzzing phase on UDP 8443 demonstrated identical results across all vantage points: every QUIC handshake attempt was dropped without server acknowledgment, proving that the blocking was protocol-based rather than port-specific.

Ultimately, the final censorship-confirmed results 150 domains in India, 104 in Malaysia, 98 in Indonesia, and 127 in Thailand represent the definitive subset of QUIC-blocked domains within the 256,951-domain test space. These consistent outcomes across independent vantage points validate the robustness of the measurement architecture and underscore the persistence of subtle, protocol-level interference mechanisms targeting QUIC traffic in multiple regional networks.

7.2 Conclusions

The experimental analysis confirms that QUIC-specific censorship is present but remains quantitatively insignificant across the examined regions as depicted on Figure 7.1. With only a minute fraction of domains affected, the results indicate that QUIC interference occurs in isolated cases rather than as part of a systematic blocking policy. The negligible proportion of censored traffic relative to the total dataset highlights that such interference does not constitute a widespread impediment to QUIC deployment. Overall, the findings suggest that regional networks largely permit QUIC communication, with only limited, selective instances of protocol-level filtering.

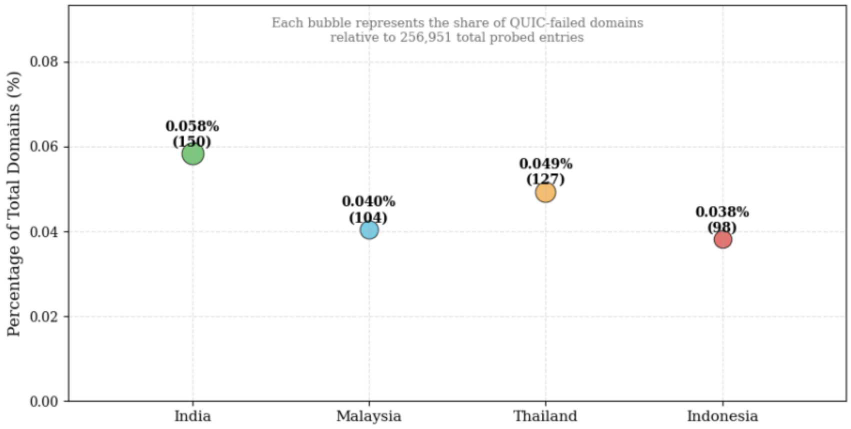


Figure 7.1: Proportion of QUIC-Specific Failures Across Regional Vantage Points

7.3 Potential Causes of QUIC Failures on VPS Vantage Points

It should also be noted that inconsistencies in QUIC connectivity do not always imply censorship. Many websites rely on geographically distributed servers or CDNs to enhance performance for users in different regions. As a result, a specific server such as one located near the Indian vantage point might not have QUIC support enabled or configured. Therefore, the lack of QUIC responses in certain regions could result from server side deployment variations rather than deliberate blocking.

In some cases, Internet Service Providers or intermediary middleboxes may implement UDP rate limiting or deep packet inspection that selectively affects QUIC traffic. Such mechanisms can unintentionally mimic censorship by dropping or delaying QUIC handshake packets

Resource limitations on t2.micro VPS instances such as CPU credit exhaustion or limited memory can lead to timeouts or incomplete handshakes. However, this factor can largely be omitted, as all tests were rerun ten times to confirm consistency and eliminate transient performance artifacts.

7.4 Potential Future Work

For future work, this thesis could be extended by conducting systematic multi-port probing to examine whether QUIC filtering is bound to specific UDP ports or represents a generalized transport-layer restriction. Additionally, controlled URL-path fuzzing and randomized query strings could be implemented to reveal application-layer or keyword-based censorship patterns beyond standard port behavior. A longitudinal measurement campaign, performed across multiple vantage points over time, would further expose temporal variations, such as time-of-day or event-driven filtering. These extensions would deepen understanding of how censorship mechanisms evolve and interact across network layers. Ultimately, such continuation would transform the current static snapshot of QUIC accessibility into a dynamic, comparative analysis of censorship persistence and adaptation.

Bibliography

[1] R. Clayton, S. J. Murdoch, and R. N. M. Watson,“ Ignoring the great firewall of China”, in Proc. Privacy Enhancing Technologies Symposium (PETS), 2006.

[2] J. Iyengar and M. Thomson, “QUIC: A UDP-Based Multiplexed and Secure Transport”, Internet Engineering Task Force (IETF) RFC 9000, May 2021. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9000

[3] Y. J. Park and J. R. Crandall, “Empirical Study of a National-Scale Distributed Intrusion Detection System: Backbone-Level Filtering of HTML Responses in China,in Proc”. USENIX Workshop on Free and Open Communications on the Internet (FOCI), 2010.

[4] S. Aryan, H. Aryan, and J. A. Halderman, “Internet Censorship in Iran:A First Look”, in Proc. 3rd USENIX Workshop on Free and Open Communications on the Internet (FOCI), 2013.

[5] R. Deibert et al., Access Denied: The Practice and Policy of Global Internet Filtering, MIT Press, 2008.

[6] A. Bock, T. H. E. Hubner, and R. Holz, “Tracking QUIC: Characterizing Ongoing Censorship of QUIC Protocol in China” in Proc. Passive and Active Measurement Conference (PAM), 2021.

[7] P. Gill, M. Crete-Nishihata, J. Dalek, S. Goldberg, and R. J. Deibert, “Characterizing Web Censorship Worldwide: Another Look at the OpenNet Initiative Data,” ACM Trans. Web, vol. 9, no. 1, pp. 1-29, 2015.

[8] B. Jones, R. Ensafi, S. Burnett, N. Feamster, and V. Paxson, “On the Challenges of Measuring Internet Censorship at Scale,” in Proc. USENIX Symposium on Networked Systems Design and Implementation (NSDI), 2020.

[9] R. Deibert, J. Palfrey, R. Rohozinski, and J. Zittrain, Access Controlled: The Shaping of Power, Rights, and Rule in Cyberspace. Cambridge, MA: MIT Press, 2010.

[10] M. E. Roberts, Censored: Distraction and Diversion Inside China’s Great Firewall. Princeton, NJ: Princeton University Press, 2018.

[11] Freedom House, “Freedom on the Net 2023: The Repressive Power of Artificial Intelligence,” 2023. [Online]. Available: https://freedomhouse.org/report/freedom-net/2023

[12] R. Deibert, Reset: Reclaiming the Internet for Civil Society. Toronto, Canada: House of Anansi, 2020.

[13] T. Tufekci, “How the Internet has made social movements more difficult to control,” Foreign Affairs, Jan./Feb. 2017.

[14] A. York, “Policing Content in the Quasi-Public Sphere,” Electronic Frontier Foundation, 2010. [Online]. Available: https://www.eff.org/deeplinks/2010/07/policing-content-quasi-public-sphere

[15] A. Gunaratne, “Sri Lanka blocks social media after Easter Sunday attacks,” BBC News, Apr. 21, 2019. [Online]. Available: https://www.bbc.com/news/world-asia-48010617

[16] S. Zuboff, The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power. New York, NY: PublicAffairs, 2019.

[17] A. T. Kenyon and M. Richardson, “New dimensions for privacy: Informational privacy and internet censorship,” International Journal of Law and Information Technology, vol. 22, no. 3, pp. 263-291, 2014.

[18] D. Fiore, J. R. Crandall, and J. Knockel, “A nuanced look at obfuscation metrics: Measuring protocol obfuscation and censorship resistance,” in Proc. 2020 Workshop on Privacy in the Electronic Society (WPES), 2020, pp. 181-194.

[19] J. R. Crandall, D. Zinn, M. Byrd, E. Barr, and R. East, “ConceptDoppler: A Weather Tracker for Internet Censorship,” in Proc. ACM Conference on Computer and Communications Security (CCS), 2007, pp. 352-365.

[20] A. Dainotti, C. Squarcella, E. Aben, K. C. Claffy, M. Chiesa, M. Russo, and A. Pescape, “Analysis of country-wide Internet outages caused by censorship,” in Proc. 2011 ACM Internet Measurement Conference (IMC), 2011, pp. 1-18.

[21] VyprVPN, ”VyprDNS | VyprVPN,” VyprVPN, [Online]. Available: label-https://www.vyprvpn.com/features/vyprdns. [Accessed: May 4, 2025].

[22] M. T. Alam, M. Z. Shafiq, and A. X. Liu, “Censorship Resistance: A Survey,” ACM Computing Surveys (CSUR), vol. 51, no. 1, pp. 1-36, 2018.

[23] J. Knockel, J. R. Crandall, and J. Saia, “Three Researchers, Five Conjectures: An Empirical Analysis of Censorship in China,” in Free and Open Communications on the Internet (FOCI), 2011.

[24] A. Houmansadr, C. Brubaker, and V. Shmatikov, “The Parrot Is Dead: Observing Unobservable Network Communications,” in IEEE Symposium on Security and Privacy, 2013, pp. 65-79.

[25] K. Elmenhorst, ”A Quick Look at QUIC Censorship,” Open Technology Fund, Apr. 20, 2022. [Online]. Available: https://www.opentech.fund/news/a-quick-look-at-quic-censorship/. [Accessed: May 4, 2025]

[26] J. F. Kurose and K. W. Ross, Computer Networking: A Top-Down Approach, 7th ed. Boston, MA: Pearson, 2016.

[27] R. Braden, "User Datagram Protocol," RFC 768, Internet Engineering Task Force, Aug. 1980. [Online]. Available: https://www.rfc-editor.org/info/rfc768

[28] Private Internet Access, "TCP vs. UDP: Understanding the Difference," Private Internet Access Blog, [Online]. Available: https://www.privateinternetaccess.com/blog/tcp-vs-udp-understanding-the-difference/. [Accessed: May 5, 2025].

[29] GeeksforGeeks, "Open Systems Interconnection Model (OSI)," GeeksforGeeks, [Online]. Available: https://www.geeksforgeeks.org/open-systems-interconnection-model-osi/. [Accessed: May 5, 2025].

[30] J. Knockel, J. R. Crandall, and J. Saia, "Three Researchers, Five Conjectures: An Empirical Analysis of Censorship in China," in Free and Open Communications on the Internet (FOCI), 2011.

[31] M. Belshe, R. Peon, and M. Thomson, "Hypertext Transfer Protocol Version 2 (HTTP/2)," RFC 7540, IETF, May 2015. [Online]. Available: https://www.rfc-editor.org/info/rfc7540

[32] D. Mishra et al., "Performance Evaluation of HTTP/3 over QUIC," in Proc. IEEE ICC 2021, Montreal, QC, Canada, 2021.

[33] G. Huston, "Comparing TCP and QUIC," APNIC Blog, Nov. 3, 2022. [Online]. Available: https://blog.apnic.net/2022/11/03/comparing-tcp-and-quic/. [Accessed: May 5, 2025].

[34] G. Huston, "A Quick Look at QUIC," APNIC Blog, Mar. 4, 2019. [Online]. Available: https://blog.apnic.net/2019/03/04/a-quick-look-at-quic/. [Accessed: May 5, 2025].

[35] Z. Weinberg et al., "A Case for a Global Censorship Observatory," in Proc. USENIX Workshop on Free and Open Communications on the Internet (FOCI), 2011.

[36] K. Elahi and I. Goldberg, "Censorship Resistance: Limits and Challenges," IEEE Internet Computing, vol. 20, no. 1, pp. 62-66, 2016.

[37] R. Sen, D. Choffnes, and A. Mislove, "Analyzing the Impact of Regional Internet Filtering," in Proc. ACM Internet Measurement Conference (IMC), 2020.

[38] OONI, How Internet censorship changed in Russia during the 1st year of military conflict in Ukraine, 2023.

[39] GitHub/net4people, Blocking of HTTP/3 (QUIC) in Russia, Issue 108, 2021.

[40] OONI, OONI measurements show ongoing internet censorship in Azerbaijan, 2023.

[41] Reporters Without Borders (RSF), Heavy Internet censorship in Kazakhstan. [Online]. Available: https://rsf.org/en/heavy-internet-censorship-kazakhstan. [Accessed: May 11, 2025]

[42] A. Dainotti et al., “Analysis of country-wide Internet outages caused by censorship,” Proc. ACM IMC, 2011.

[43] D. Fiore, J. R. Crandall, and J. Knockel, “A nuanced look at obfuscation metrics,” WPES, 2020.

[44] K. V. S. Rao, HTTK: Characterizing and Evading Application-Layer Censorship by TransTeleKom, University of Maryland Honors Thesis, 2020.

[45] M. VanderSloot et al., “Towards a Censorship Analyzer for Network Traffic,” USENIX Security Symposium, 2018.

[46] S. Wendzel et al., ”A Survey of Internet Censorship and its Measurement: Methodology, Trends, and Challenges,” arXiv preprint arXiv:2502.14945, 2025.

[47] K. Elmenhorst, B. Schütz, N. Aschenbruck, and S. Basso, “Web Censorship Measurements of HTTP/3 over QUIC,” in Proc. ACM Internet Measurement Conference (IMC), 2021. [Online]. Available: https://dl.acm.org/doi/10.1145/3487552.3487836

[48] M. Le Pochat, T. Van Goethem, S. Tajalizadehkhoob, M. Korczynski, and W. Joosen, “Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation,” in Proc. of the 26th Annual Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 2019. [Online]. Available: https://tranco-list.eu. Accessed: May 25, 2025.

[49] Fortinet, FortiGuard Web Filter Lookup, Available: https://www.fortiguard.com/webfilter. Accessed: May 25, 2025.

[50] Webshrinker, Domain Category API, Available: https://www.webshrinker.com. Accessed: May 25, 2025.

[51] Firebog.net, Curated Blocklists for DNS Filtering, Available: https://firebog.net. Accessed: May 25, 2025.

[52] Freedom House, Freedom in the World 2024 Switzerland. [Online]. Available: https://freedomhouse.org/country/switzerland/freedom-world/2024. [Accessed: May 29, 2025].

[53] Swiss Gambling Supervisory Authority (Gespa), “Access blocking,” [Online]. Available: https://www.gespa.ch/en/fighting-illegal-gambling/access-blocking. [Accessed: May 29, 2025].

[54] W3Techs, “Usage of HTTP/3 for websites,” W3Techs World Wide Web Technology Surveys. [Online]. Available: https://w3techs.com/technologies/details/ce-http3. [Accessed: May 11, 2025].

[55] K. Elmenhorst, B. Schütz, N. Aschenbruck, S. Basso, et al., “Web censorship measurements of HTTP/3 over QUIC,” in Proc. Internet Measurement Conference (IMC), 2021. [Online]. Available: https://dl.acm.org/doi/10.1145/3487552.3487836 . [Accessed: September 28, 2025].

[56] OONI, "Measuring HTTP/3 censorship with OONI Probe," OONI, Jan. 2022. [Online]. Available: https://ooni.org/post/2022-http3-measurements-paper/ . [Accessed: September 28, 2025].

[57] A. Zohaib, Q. Zao, J. Sippe, et al., "Exposing and circumventing SNI-based QUIC censorship of the Great Firewall of China," in Proc. USENIX Security Symposium, 2025. [Online]. Available: https://www.usenix.org/system/files/usenixsecurity25-zohaib.pdf . [Accessed: September 28, 2025].

[58] J. Sengupta and V. Bajpai, "On cross-layer interactions of QUIC, encrypted DNS and HTTP/3," in IEEE Trans. Network and Service Management (TNSM), 2024. [Online]. Available: https://vaibhavbajpai.com/documents/papers/proceedings/quic-tnsm-2024.pdf . [Accessed: September 28, 2025].

[59] S. Wendzel, et al., "A survey of Internet censorship and its measurement: Methodology, trends, and challenges," arXiv preprint, Feb. 2025. [Online]. Available: https://arxiv.org/abs/2502.14945 . [Accessed: September 28, 2025].

[60] M. Isah, A. Phokeer, J. Chavula, A. Elmokashfi, and A. S. Asrese, "State of Internet measurement in Africa, a survey," AFRICOMM/Aalto University Report, 2020. [Online]. Available: https://amreesh.github.io/assets/pdf/2020-africomm-internet-measurement-africa.pdf . [Accessed: September 28, 2025]

[61] APNIC Foundation, "Measuring and detecting network interference in Southeast Asia," APNIC Foundation Technical Report, 2023. [Online]. Available: https://apnic.foundation/projects/measuring-and-detecting-network-interference-in-southeast-asia/technicalreport/ . [Accessed: September 28, 2025].

[62] A. Bhaskar, A. Pearce, S. Iyer, S. Even, et al., "Understanding routing-induced censorship changes via Monocle," in Proc. ACM CCS, 2024. [Online].

[63] A. A. Niaki, et al., "ICLab: A global, longitudinal Internet censorship measurement platform," in Proc. Internet Measurement Conference (IMC), 2020. [Online]. .[Accessed: September 28, 2025].

[64] M. Ivanovic, F. Wirz, J. Subira Nieto, and A. Perrig, "Charting censorship resilience and global Internet reachability: A quantitative approach," arXiv preprint, 2024. [Online]. Available: https://arxiv.org/abs/2403.09447 . [Accessed: September 28, 2025]

Abbreviations

DPI	Deep Packet Inspection
DNS	Domain Name System
IP	Internet Protocol
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
TCP	Transmission Control Protocol
QUIC	Quick UDP Internet Connections
IETF	Internet Engineering Task Force
TLS	Transport Layer Security
UDP	User Datagram Protocol
HTTP/3	Hypertext Transfer Protocol Version 3
VPS	Virtual Private Server
URL	Uniform Resource Locator
VoIP	Voice over Internet Protocol
RFC	Request for Comments
HOL	Head-of-Line
0RTT	Zero Round Trip Time
GDPR	General Data Protection Regulation
HTML	HyperText Markup Language
IDS	Intrusion Detection System
ISP	Internet Service Provider
ICLab	Information Controls Lab
RSF	Reporters Sans Frontiëres
API	Application Programming Interface
CSV	Comma-Separated Values
JSON	JavaScript Object Notation
Gespa	Glücksspielaufsichtsbehörde
ALPN	Application-Layer Protocol Negotiation
CLI	Elastic Compute Cloud
AWS	Amazon Web Services
SSH	Secure Shell
VPN	Virtual Private Network
FPR	False Positive Rate
CDN	Content Delivery Network
SNI	Server Name Indication
IKEv2	Internet Key Exchange version 2

OONI	Open Observatory of Network Interference
AS	Autonomous System
Alt-Svc	Alternative Service (HTTP/3 header extension for QUIC/HTTP/3)
EC2	Elastic Compute Cloud (Amazon AWS service for VPS)

Glossary

Censorship Device A network appliance or system deployed at the national or institutional perimeter that inspects, filters, or blocks internet traffic according to policy [5].

DPI A technique where network equipment analyzes the content of packets (beyond headers) to identify applications, protocols, or keywords for monitoring, blocking, or throttling [1].

DNS Manipulation A censorship method that interferes with the Domain Name System, for example by returning false IP addresses or error responses to prevent users from reaching specific websites [3].

Great Firewall of China The nickname for China's highly complex internet censorship system, which combines multiple techniques to control access to global internet content [1].

HTTP The application-layer protocol used for transmitting web pages and related resources [27].

HTTPS A secure version of HTTP that uses TLS encryption to protect communications [27].

IETF An international standards organization that develops and maintains internet protocols such as TCP, TLS, and QUIC [2].

IP Address Blocking A censorship method where access to certain servers or services is denied by blocking their associated IP addresses [4].

Metadata Data that provides information about other data, such as headers or control information in packets. QUIC encrypts much of this metadata, limiting visibility to censors [2].

Protocol Fingerprinting A censorship technique that identifies traffic by analyzing unique characteristics (e.g., packet sizes, handshake patterns) associated with specific protocols [18].

QUIC A transport protocol developed by Google and standardized by the IETF. It runs over UDP, integrates TLS from the start, encrypts more metadata than TCP, and powers HTTP/3 [2].

TCP A connection-oriented transport protocol widely used for reliable communication on the internet [27].

TLS A cryptographic protocol that secures internet communications by providing confidentiality, integrity, and authentication [2].

VPS A virtual machine provided by a hosting company or cloud provider. In this thesis, VPSs serve as remote vantage points for conducting censorship measurements [62].

UDP A lightweight, connectionless transport protocol. QUIC is built on top of UDP [28].

Application-Layer Censorship Censorship that targets higher-level protocols by inspecting or manipulating metadata, headers, or payloads rather than simply blocking traffic at the transport layer [6].

CDN A distributed network of servers that delivers web content closer to users to improve performance. CDNs complicate IP-based censorship because many services share the same IP address ranges [34].

Connection Migration A QUIC feature allowing ongoing connections to continue seamlessly even if the clientâs IP address changes [2].

DNSSEC A set of security protocols that add authentication to DNS responses, reducing the risk of DNS manipulation [28].

Encrypted DNS Secure versions of DNS that encrypt queries (DNS over HTTPS, DNS over TLS), making censorship by DNS interference more difficult [59].

False Positive In censorship detection, a case where legitimate traffic is incorrectly flagged or blocked as censored traffic [18].

Head-of-Line Blocking A limitation of TCP where a single lost packet delays all subsequent packets in the same connection, impacting performance in HTTP/2. QUIC avoids this issue [33].

Keyword Filtering Censorship method where DPI devices scan traffic for predefined keywords and block or reset connections when detected [19].

Multiplexing A feature in QUIC and HTTP/2/3 that allows multiple independent streams of data to be transmitted simultaneously over a single connection without interference [32].

OpenVPN / IKEv2 VPN protocols commonly used for encrypted tunneling [62].

Port-Based Filtering A censorship method that blocks traffic by restricting access to known port numbers [4].

Protocol Obfuscation Techniques used to disguise traffic patterns so that censors cannot easily identify the protocol [18].

SNI A TLS extension that reveals the hostname a client is connecting to during the handshake. Often exploited by censors for domain-based filtering [58].

TCP Reset Injection A censorship technique where forged TCP reset packets are sent to terminate a connection, often used when prohibited keywords or domains are detected [19].

TLS 1.3 The latest version of the TLS encryption protocol, offering stronger privacy and integrated by default into QUIC to reduce metadata visibility [2].

Transport-Layer Censorship Blocking or interfering at the TCP/UDP layer, such as IP address blocking, port filtering, or UDP packet dropping [6].

Variable-Length Headers A QUIC feature where packet headers vary in size, making it more difficult for censors to create static detection rules [2].

Active Probing Sending test requests from measurement vantage points to detect censorship by comparing expected vs. observed responses [20].

AS A collection of IP networks under a single administrative control [27].

Backbone-Level Filtering Censorship applied at the core of a countryâs internet infrastructure [3].

ConceptDoppler An early censorship detection tool that monitored keyword-based filtering by analyzing response anomalies [19].

Handshake Fingerprinting Identifying a protocol by analyzing the characteristics of its handshake messages [49].

ICLab A censorship measurement platform focusing on large-scale global studies [64].

Longitudinal Studies Measurements conducted repeatedly over time to observe how censorship behaviors change during events [61].

OONI A widely used open-source project that measures censorship worldwide using distributed probes [39].

Baseline A reference measurement location (Switzerland in this study) assumed to be free of censorship, used for comparison with test sites [54].

Blocklist A list of domains or IPs that are intentionally restricted (e.g., Gespa gambling blacklist, Firebog adult-content lists) [53].

Curl (with HTTP/3) A command-line tool used for sending HTTP requests [35].

Dataset Curation The process of cleaning and filtering domain lists to ensure ethical and reliable measurements [50].

Domain Categorization Tools Services like FortiGuard, Webshrinker, and Firebog used to classify or block domains based on content type [51].

Gespa The regulatory body in Switzerland that enforces online gambling restrictions via blacklists [55].

HTTP Status Code Standard response codes returned by servers (e.g., 200 OK, 403 Forbidden), used in this study to evaluate probe results [27].

Measurement Workflow The step-by-step logic applied during probing: test with QUIC, fallback to TCP/TLS if QUIC fails, log results [49].

Module A functional component of the measurement tool (e.g., Input Handler, QUIC Probe, TCP/TLS Fallback, Result Logger, Rate Limiter) [46].

Rate Limiter A mechanism in the probing tool that introduces short delays between requests to avoid overwhelming servers and remain ethical [8].

Tranco List A reproducible ranking of the top 1 million websites, used as the starting dataset for selecting QUIC-enabled domains [50].

Vantage Point A remote server (VPS) deployed in a target country to conduct censorship measurements from that location [62].

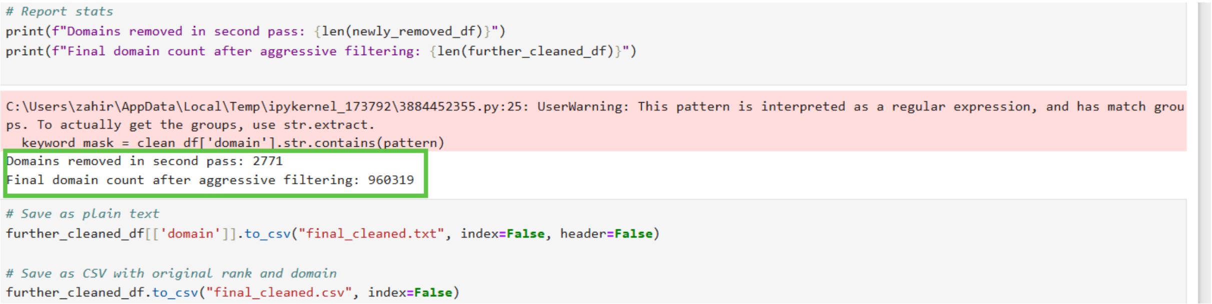
List of Figures

List of Tables

Appendix A

Scripts and Resources Used for Experiment

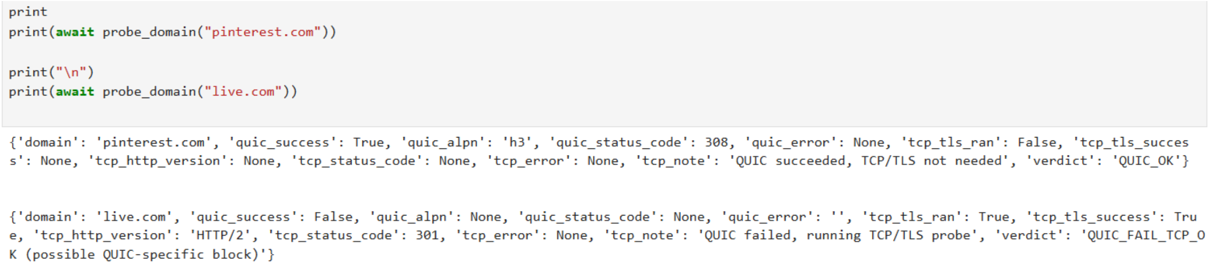
A.1 Filtering Scripts



```
# Report stats
print(f"Domains removed in second pass: {len(newly_removed_df)}")
print(f"Final domain count after aggressive filtering: {len(further_cleaned_df)}")

C:\Users\zahir\AppData\Local\Temp\ipykernel_173792\3884452355.py:25: UserWarning: This pattern is interpreted as a regular expression, and has match grou
ps. To actually get the groups, use str.extract.
  keyword_mask = clean_df['domain'].str.contains(pattern)
Domains removed in second pass: 2771
Final domain count after aggressive filtering: 960319

# Save as plain text
further_cleaned_df[['domain']].to_csv("final_cleaned.txt", index=False, header=False)

# Save as CSV with original rank and domain
further_cleaned_df.to_csv("final_cleaned.csv", index=False)
```

Figure A.1: Initial Domain Cleanup

Figure A.1 script performs a final round of aggressive filtering by removing domains matching unwanted keyword patterns to refine the dataset. It then reports the number of domains removed (2,771) and the final count retained after cleaning (960,319).



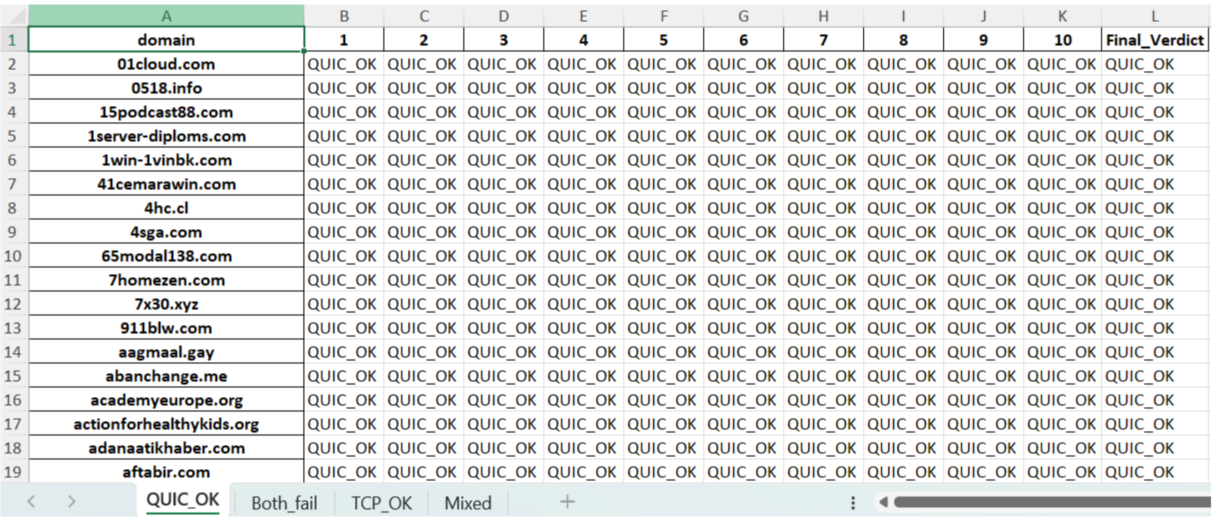
```
print
print(await probe_domain("pinterest.com"))

print("\n")
print(await probe_domain("live.com"))

{'domain': 'pinterest.com', 'quic_success': True, 'quic_alpn': 'h3', 'quic_status_code': 308, 'quic_error': None, 'tcp_tls_ran': False, 'tcp_tls_succes
s': None, 'tcp_http_version': None, 'tcp_status_code': None, 'tcp_error': None, 'tcp_note': 'QUIC succeeded, TCP/TLS not needed', 'verdict': 'QUIC_OK'}


{'domain': 'live.com', 'quic_success': False, 'quic_alpn': None, 'quic_status_code': None, 'quic_error': '', 'tcp_tls_ran': True, 'tcp_tls_success': Tru
e, 'tcp_http_version': 'HTTP/2', 'tcp_status_code': 301, 'tcp_error': None, 'tcp_note': 'QUIC failed, running TCP/TLS probe', 'verdict': 'QUIC_FAIL_TCP_O
K (possible QUIC-specific block)'}
```

Figure A.2: Sanity Check on Varied Domains on Output

Figure A.2 Sanity check demonstrates the tool's ability to differentiate between QUIC-supported and non-QUIC domains. The probe shows that pinterest.com successfully

completes a QUIC handshake, while live.com fails over to TCP/TLS, indicating a possible QUIC-specific block.

A.2 Representative Output from a VPS over 10 Test Runs

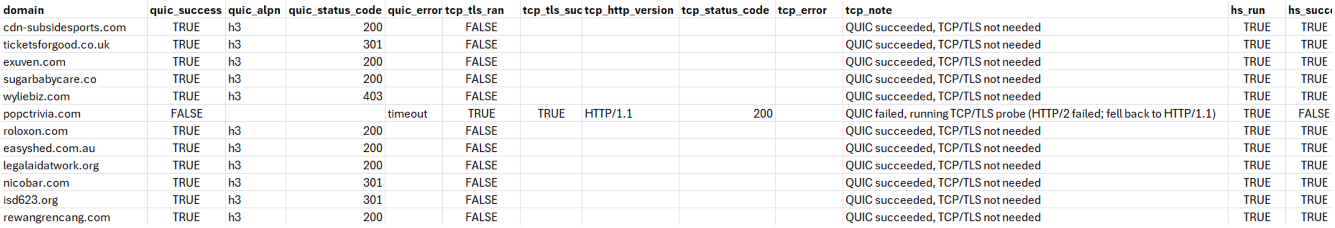


	A	B	C	D	E	F	G	H	I	J	K	L
1	domain	1	2	3	4	5	6	7	8	9	10	Final_Verdict
2	01cloud.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
3	0518.info	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
4	15podcast88.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
5	1server-diploms.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
6	1win-1vinbk.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
7	41cemarawin.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
8	4hc.cl	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
9	4sga.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
10	65modal138.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
11	7homezen.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
12	7x30.xyz	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
13	911blw.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
14	aagmaal.gay	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
15	abanchange.me	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
16	academyeurope.org	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
17	actionforhealthykids.org	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
18	adanaatikhaber.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK
19	aftabir.com	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK	QUIC_OK

Figure A.3: Results after Probing 10 Runs on Failed Domains

Figure A.3 Presents the revalidation results of previously QUIC-failed domains, each reprobed ten times to ensure consistency in outcomes. The domains were then classified into four categories QUIC_OK, Both_Fail, TCP_OK, and Mixed based on their final verdict after repeated testing.

A.3 Excerpt of a Result from the Tool



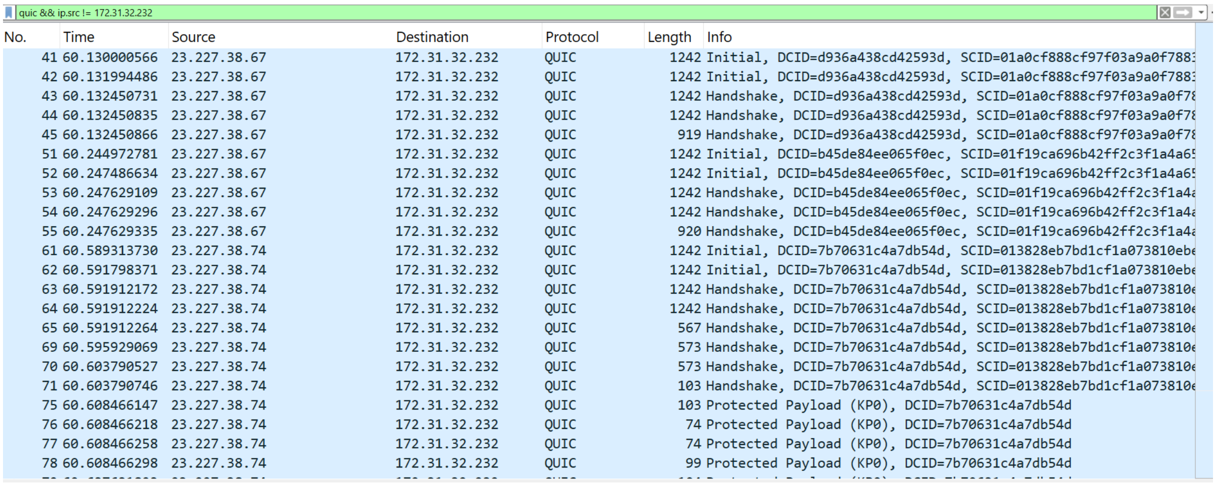
domain	quic_success	quic_alpn	quic_status_code	quic_error	tcp_tls_ran	tcp_tls_suc	tcp_http_version	tcp_status_code	tcp_error	tcp_note	hs_run	hs_succ
cdn-subsidesports.com	TRUE	h3	200		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE
ticketsforgood.co.uk	TRUE	h3	301		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE
exuven.com	TRUE	h3	200		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE
sugarbabycare.co	TRUE	h3	200		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE
wyliebiz.com	TRUE	h3	403		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE
popctrivia.com	FALSE			timeout	TRUE	TRUE	HTTP/1.1	200		QUIC failed, running TCP/TLS probe (HTTP/2 failed; fell back to HTTP/1.1)	TRUE	FALSE
roloxon.com	TRUE	h3	200		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE
easyshed.com.au	TRUE	h3	200		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE
legalaidatwork.org	TRUE	h3	200		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE
nicobar.com	TRUE	h3	301		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE
isd623.org	TRUE	h3	301		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE
rewangrencang.com	TRUE	h3	200		FALSE					QUIC succeeded, TCP/TLS not needed	TRUE	TRUE

Figure A.4: Actual Run from a VPS

A.4 Sample Packet Capture on QUIC Failed Probes

Figure A.4 Shows the actual probing results generated by the measurement tool, capturing both QUIC and TCP/TLS connection outcomes for each domain. It records detailed

parameters such as QUIC success, ALPN negotiation, status codes, and fallback behavior, illustrating how the tool differentiates successful QUIC handshakes from TCP fallbacks or timeouts during real network measurements.



```
quic && ip.src != 172.31.32.232
No.   Time         Source          Destination     Protocol  Length Info
   41 60.130000566 23.227.38.67    172.31.32.232   QUIC        1242 Initial, DCID=d936a438cd42593d, SCID=01a0cf888cf97f03a9a0f7883
   42 60.131994486 23.227.38.67    172.31.32.232   QUIC        1242 Initial, DCID=d936a438cd42593d, SCID=01a0cf888cf97f03a9a0f7883
   43 60.132450731 23.227.38.67    172.31.32.232   QUIC        1242 Handshake, DCID=d936a438cd42593d, SCID=01a0cf888cf97f03a9a0f78
   44 60.132450835 23.227.38.67    172.31.32.232   QUIC        1242 Handshake, DCID=d936a438cd42593d, SCID=01a0cf888cf97f03a9a0f78
   45 60.132450866 23.227.38.67    172.31.32.232   QUIC         919 Handshake, DCID=d936a438cd42593d, SCID=01a0cf888cf97f03a9a0f78
   51 60.244972781 23.227.38.67    172.31.32.232   QUIC        1242 Initial, DCID=b45de84ee065f0ec, SCID=01f19ca696b42ff2c3f1a4a65
   52 60.247486634 23.227.38.67    172.31.32.232   QUIC        1242 Initial, DCID=b45de84ee065f0ec, SCID=01f19ca696b42ff2c3f1a4a65
   53 60.247629109 23.227.38.67    172.31.32.232   QUIC        1242 Handshake, DCID=b45de84ee065f0ec, SCID=01f19ca696b42ff2c3f1a4a
   54 60.247629296 23.227.38.67    172.31.32.232   QUIC        1242 Handshake, DCID=b45de84ee065f0ec, SCID=01f19ca696b42ff2c3f1a4a
   55 60.247629335 23.227.38.67    172.31.32.232   QUIC         920 Handshake, DCID=b45de84ee065f0ec, SCID=01f19ca696b42ff2c3f1a4a
   61 60.589313730 23.227.38.74    172.31.32.232   QUIC        1242 Initial, DCID=7b70631c4a7db54d, SCID=013828eb7bd1cf1a073810ebe
   62 60.591798371 23.227.38.74    172.31.32.232   QUIC        1242 Initial, DCID=7b70631c4a7db54d, SCID=013828eb7bd1cf1a073810ebe
   63 60.591912172 23.227.38.74    172.31.32.232   QUIC        1242 Handshake, DCID=7b70631c4a7db54d, SCID=013828eb7bd1cf1a073810e
   64 60.591912224 23.227.38.74    172.31.32.232   QUIC        1242 Handshake, DCID=7b70631c4a7db54d, SCID=013828eb7bd1cf1a073810e
   65 60.591912264 23.227.38.74    172.31.32.232   QUIC         567 Handshake, DCID=7b70631c4a7db54d, SCID=013828eb7bd1cf1a073810e
   69 60.595929069 23.227.38.74    172.31.32.232   QUIC         573 Handshake, DCID=7b70631c4a7db54d, SCID=013828eb7bd1cf1a073810e
   70 60.603790527 23.227.38.74    172.31.32.232   QUIC         573 Handshake, DCID=7b70631c4a7db54d, SCID=013828eb7bd1cf1a073810e
   71 60.603790746 23.227.38.74    172.31.32.232   QUIC         103 Handshake, DCID=7b70631c4a7db54d, SCID=013828eb7bd1cf1a073810e
   75 60.608466147 23.227.38.74    172.31.32.232   QUIC         103 Protected Payload (KP0), DCID=7b70631c4a7db54d
   76 60.608466218 23.227.38.74    172.31.32.232   QUIC          74 Protected Payload (KP0), DCID=7b70631c4a7db54d
   77 60.608466258 23.227.38.74    172.31.32.232   QUIC          74 Protected Payload (KP0), DCID=7b70631c4a7db54d
   78 60.608466298 23.227.38.74    172.31.32.232   QUIC          99 Protected Payload (KP0), DCID=7b70631c4a7db54d
```

Figure A.5: Wireshark Packet Capture

Figure A.3 Wireshark capture illustrates QUIC traffic observed from the Malaysian vantage point, filtered to display only packets sent from servers to the VPS client (172.31.32.232). It was used to verify QUIC handshake responses for domains that previously failed in the probing stage, confirming which servers actively responded to the clientâs initial QUIC requests.