GSC Advanced Research and Reviews

eISSN: 2582-4597 CODEN (USA): GARRC2

Cross Ref DOI: 10.30574/gscarr

Journal homepage: https://gsconlinepress.com/journals/gscarr/

(Review Article)

An investigation of TCP security and performance issues when deployed in high-speed networks

Patience Mmasi Robai *

Jaramogi Oginga Odinga University of Science and Technology, Bondo, Kenya.

GSC Advanced Research and Reviews, 2024, 21(02), 282–314

Publication history: Received on 03 October 2024; revised on 14 November 2024; accepted on 16 November 2024

Article DOI: https://doi.org/10.30574/gscarr.2024.21.2.0130

Abstract

As high-speed networks become more prevalent in modern communication infrastructures, the security and performance of the underlying transport protocols become increasingly important. A few particular problems arise when Transport Control Protocol (TCP), a crucial component of the internet, is implemented on high-speed networks. This study delves deeply into the performance and security concerns related to TCP in high-speed networks, including the effects of higher data rates and shorter round trip durations. It also suggests possible remedies that could lessen these risks. The study investigates the inefficiencies in TCP's congestion control algorithms, which are optimized for slower networks, when it comes to making the most use of available bandwidth in high-speed settings. It looks at the effects of being more susceptible to Distributed Denial of Service (DDoS) assaults, which have the ability to overload TCP connections and undermine their dependability. The paper examines current TCP variants and additions, such as TCP BBR and TCP SACK, which attempt to increase TCP's adaptability to high-speed network situations, in order to address these issues. It assesses how well these solutions work to mitigate the performance difficulties that have been identified and to sustain dependable and effective communication in high-speed contexts. The primary goals of the research are to improve our knowledge of TCP security and performance in high-speed networks and to aid in the development of more robust and secure communication protocols.

Keywords*:* TCP Performance Issues; TCP Security Issues; TCP BBR; TCP SACK; TCP FACK; RTT; BDP; Packet Reordering; Sequence Prediction.

1. Introduction

With their ultra-fast data transmission speeds, high-speed networks are now essential for handling bandwidth-demanding services and applications. Nevertheless, the implementation of conventional protocols, like TCP, in these settings has security and performance issues that require careful analysis [1]-[5]. These problems include DDoS attacks [6], latency, and packet loss Attacks using TCP Sequence Prediction and congestion worsen, affecting the general security and dependability of data transfer. In our linked society, the spread of high-speed networks has completely changed how information is accessed and transmitted. The Transmission Control Protocol (TCP), which serves as the foundation for internet connection, has been essential in guaranteeing dependable and well-organized data transmission. On the other hand, the high-speed networks that define today's digital environment were not around when the architectural underpinnings of TCP were developed [7]. The inability of TCP's congestion control techniques to fully utilize high-speed networks is one of the main causes for concern.

Because they were created for lower bandwidths, the traditional algorithms frequently find it difficult to quickly adjust to the higher data rates, which results in underutilization of the network's resources. The study also examines TCP's increased susceptibility to Distributed Denial of Service (DDoS) assaults on high-speed networks [8], illuminating the

* Corresponding author: Patience Mmasi Robai.

consequences of network saturation, congestion, and possible interruptions to TCP connections' dependability. Another set of difficulties is packet reordering and duplication, which are more common in high-speed networks due to the greater data transfer rates [9]-[11]. Out-of-order delivery and duplicate packets are discussed in detail, along with the influence on TCP's throughput, latency [12], and overall performance. This research paper will examine current TCP variations and changes intended to optimize TCP for high-speed contexts in order to overcome these issues. Investigating protocols like TCP BBR and TCP SACK will be essential to assessing how well they mitigate the performance problems that have been found, guaranteeing that TCP will continue to be a reliable and flexible protocol in the era of high-speed networking.

This research attempts to offer insights that contribute to TCP's improvement for smooth integration and long-term efficiency in high-speed networks by a thorough analysis of these performance concerns and hence contributing to the improvement of TCP performance and flexibility in the dynamic world of networking. This paper also addresses TCP's increased vulnerability to sophisticated attacks, especially those that take advantage of high-bandwidth communication's quick speed. In the context of high-speed data transmission, the study examines how well traditional security techniques work to counter risks including man-in-the-middle attacks, session hijacking, and packet interception [13]-[17]. The response time for identifying and addressing security problems becomes crucial as network speeds rise, requiring a reassessment of TCP's resistance to various cyber threats. The study also looks at how Distributed Denial of Service (DDoS) assaults, a common threat in high-speed networks, might take advantage of TCP's special features to overwhelm systems and perhaps compromise data. The study looks into how TCP's congestion control algorithms interact with high-speed DDoS attacks to give light on the threats to network availability and the efficacy of solutions. Encryption is another area in which TCP and security in high-speed networks are intricately connected [18]-[23]. The effectiveness of encryption protocols, such TLS (Transport Layer Security), is evaluated for its capacity to preserve the confidentiality and integrity of transmitted data as data travels across networks at previously unheard-of speeds. This research study will evaluate current security protocols and suggest improvements designed for high-speed TCP installations in order to navigate these security problems. The goal is to present a comprehensive picture of the security environment, with insights that help design tactics and protocols that can protect TCP from new attacks in high-speed networking settings.

2. TCP security issues when deployed in high-speed networks

The concept of security is highly connected to the need for protecting sensitive data from unauthorized access. With the growing use of internet infrastructure for commercial applications, the demand for Quality of Service has increased at a very high rate [24]-[28]. An increasing number of application need complex, reliable control protocols for guaranteeing the Quality of Service. The TCP/IP suite has many design weaknesses as far as security is concerned, probably because when the development took place (1970) network attacks were almost unknown [29]. These weaknesses usually present a number of problematic issues which shall be discussed in this paper. Also, we shall look at the proposed solutions to the same issues, so as to help mitigate their impact on the networks.

2.1. TCP Sequence Prediction Attacks

The TCP Sequence Prediction attacks usually involve an attacker predicting the sequence numbers of TCP packets to hijack or manipulate a communication session. In the high-speed networks, the impact of these attacks can be particularly significant due to the increased volume of data and the rapid pace at which the packets are transmitted [30], [31]. It basically indicates that the packets that were delivered were not formed by the authorized host; rather, they were created by a third party. How are they going to pull this off? Sending packets with the same source IP address is one method of doing this, which involves listening in on a discussion between two trustworthy hosts. The attacker can thus obtain the sequential number that is then used along with the IP address to send the fake packets before the legitimate host does by keeping an eye on the traffic prior to the attack [32].

The TCP Sequence Prediction attacks affect the security of TCP in high-speed networks in the following ways:

Sequence number predictability – In TCP, the sequence numbers are used to order and reassemble packets at the receiving end. An attacker can insert harmful material into the communication stream or most likely alter the connection if they can correctly forecast the sequence numbers [33].

Session hijacking and injection – By inserting malicious packets [34] into the communication stream, an attacker can take control of a TCP session with precise sequence number prediction. Due to the volume of data being transferred across high-speed networks, attackers can more easily blend in their injected packets, making it more challenging to identify and stop such attacks. As shown in Figure 1, the session hijacking attack compromises the session token by stealing or predicting a valid session token to gain unauthorized access to the Web Server [35]-[38]. The session token could be compromised in ways such predictable session token, session sniffing, client-side attacks, man-in-the-middle (MitM) attacks [39]. In the diagram below, the first attacker uses a sniffer to capture a valid token session called "Session ID", then they use the valid token session to gain authorized access to the web server.

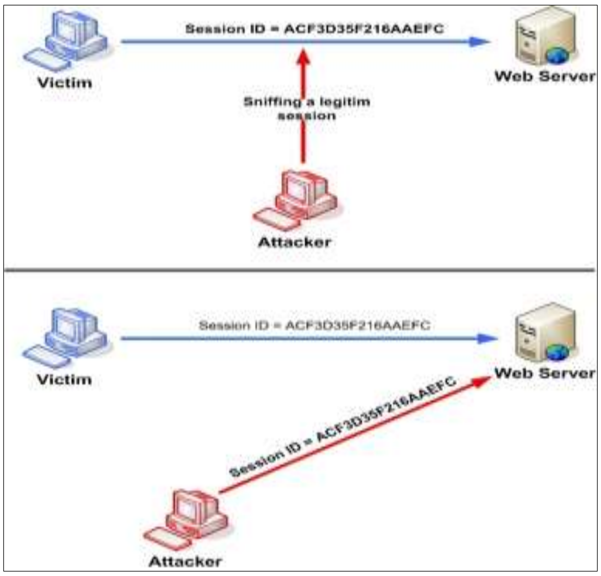

Victim
Session ID = ACF3D35F216AAEFC
Web Server
Sniffing a legitim session
Attacker
Victim
Session ID = ACF3D35F216AAEFC
Web Server
Session ID = ACF3D35F216AAEFC
Attacker

Figure 1 Manipulating the token session executing the session hijacking attack

Denial-of-Service attacks - Through the deployment of malicious reset (RST) packets, which break existing connections, TCP Sequence prediction attacks can be used to impede communication [40], [41]. An attacker can send out more of these malicious packets more quickly via high-speed networks, which could result in more severe and destructive denial-of-service attacks.

Resource exhaustion- A high volume of malicious packets may be sent across the network in an attempt to disrupt communication [44], [45]. This can result in resource depletion on the network, infrastructure, and targeted systems in high-speed networks [46]. Malicious packets can spread quickly and overload switches, routers, and other network equipment, negatively affecting their availability and functionality.

2.2. Increased Vulnerability to DDoS Attacks

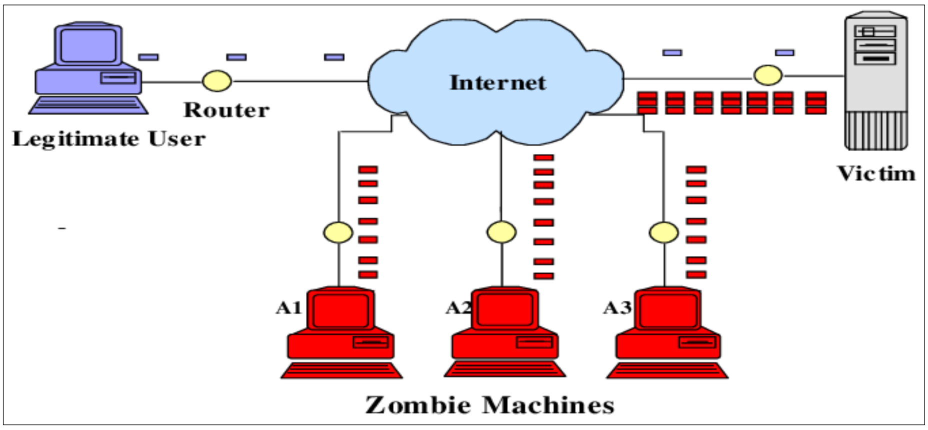


Figure 2 Illustration of DDoS Attack scenario

The higher throughput of high-speed networks makes them more susceptible to Distributed Denial of Service (DDoS) attacks, overwhelming TCP connections and causing service disruptions [46]-[52]. Increased vulnerability to Distributed Denial of Service (DDoS) attacks can have several implications for TCP in high-speed networks. As shown in Figure 2, DDoS attacks aim to overwhelm a network, service, or application with a flood of traffic, disrupting normal operations.

Increased Vulnerability to DDoS attacks influences the security of TCP when deployed in high speed networks in the following ways:

Resource exhaustion: DDoS attacks produce enormous amounts of malicious traffic directed at particular network resources. The sheer amount of attack traffic in high-speed networks can quickly deplete memory, processing power, and bandwidth [53]-[57]. Resource limitations can affect TCP connections, resulting in decreased performance and perhaps generating issues for authorized users.

Congestion packet loss: DDoS attacks have the ability to overload a network with traffic, causing congestion. This congestion may be seen by TCP as an indication of network stress because it depends on congestion management methods to adjust to network conditions [58]-[63]. In response, TCP's congestion control mechanisms might shrink the congestion window, which would lower throughput. Congestion can also cause an increase in packet loss, which can set off TCP's congestion avoidance algorithms and negatively affect performance even more [64].

Increased Latency: A DDoS attack's congestion can significantly increase network latency [65], [66]. The round-trip duration of TCP connections may be impacted by this latency, which could slow down data transfer rates and have an effect on real-time applications.

Connection establishment challenges and difficulties in distinguishing the legitimate traffic: TCP connections rely on the successful completion of the three-way handshake for proper establishment. DDoS attacks can flood the network with connection requests, making it challenging for legitimate TCP connections to establish and maintain [67], [68]. High-speed DDoS attacks often involve a mix of legitimate and malicious traffic. Distinguishing between the two becomes more challenging in high-speed networks, making it difficult for network defenses to filter out malicious packets effectively. TCP may experience false positives in congestion control, leading to inefficient utilization of available network resources [69].

Impact on network availability: DDoS attacks have the potential to overwhelm network links to the point that genuine traffic cannot use them [70], [71]. This may lead to network outages and unavailability, which would impact TCP-dependent services and apps.

3. Inefficient Congestion Control

Traditional TCP congestion control mechanisms may not scale effectively in high-speed networks, leading to suboptimal performance and potential security vulnerabilities [72], [73]. Inefficient congestion control in TCP can significantly impact its performance when deployed in high-speed networks. TCP is a widely used transport layer protocol [74] that is designed to provide reliable and ordered delivery of data over a network. However, its original congestion control mechanisms were developed in an era when network speeds were much lower compared to today's high-speed networks. Figure 3 shows the dynamics of TCP congestion control mechanisms.

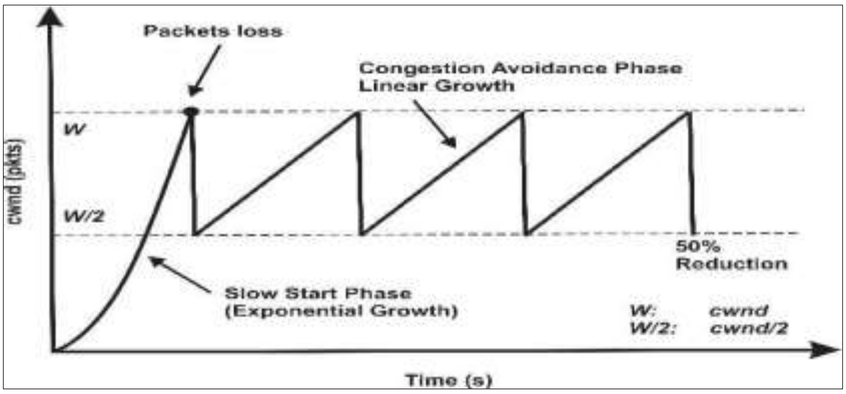


Figure 3 Dynamics of TCP Congestion Control

Inefficient Congestion Control mechanisms affect the security of TCP when deployed in high speed networks in the following ways:

Underutilization of network capacity: TCP's congestion control techniques were primarily developed for networks with smaller bandwidths. On high-speed networks, these algorithms might not be able to efficiently utilize all of the available bandwidth [75]-[78]. Slow start times and conservative congestion avoidance technologies that take longer to ramp up the congestion window might lead to underutilization of the network's capacity.

Long queues and increased latency: Longer lineups might result from network switches and routers that lack efficient congestion control [79]. This greater queuing delay leads to higher round-trip durations and latency [80], which can negatively impact the overall performance of an application. For high-speed networks to prevent excessive delay and queuing, more sophisticated congestion control algorithms are required.

Difficulty in detecting congestion: Packet loss serves as an implicit indicator of congestion for TCP's congestion management mechanism [81, [82]. Nevertheless, a packet's round-trip time across a high-speed network is shorter, which makes it difficult for TCP to reliably identify congestion based alone on packet loss. False congestion signals can cause rapid retransmissions, which can result in inefficient use of network resources.

Inefficient window size adjustments: TCP's window size adjustments may be too cautious for high-speed networks. Inadequate data transfer speeds may arise from the slow increase of the congestion window relative to the available bandwidth [83], [84]. Achieving high throughput in high-speed networks [85] necessitates efficient window size adjustments.

Unfairness in bandwidth allocation: In instances when several TCP connections share a single network, inadequate congestion control solutions may lead to unfair bandwidth distribution [86], [87]. If some connections are starved while others consume all of the available bandwidth, this could affect the network's overall fairness.

Limited scalability: Inefficient congestion control algorithms may experience scalability problems in high-speed networks with a large number of connections [88], [89]. The overhead of managing congestion across numerous connections is one such barrier.

4. Packet Reordering and Duplication

The rapid transmission of packets in high-speed networks can result in packet reordering and duplication [90], challenging the standard TCP sequence number verification mechanisms. Packet reordering and duplication can have notable effects on TCP when deployed in high-speed networks. TCP is designed to provide reliable, in-order delivery of data, and variations from this order can impact performance [91]. In high-speed networks, the likelihood of encountering packet reordering or duplication can increase due to factors like parallel processing, multiple paths, and the inherent complexity of high-speed environments [92].

Packet reordering and duplication influences the performance of TCP when deployed in high-speed networks in the following ways:

Impact on throughput: In TCP, packet reordering may result in needless retransmissions. TCP reduces the effective throughput [93] when it detects packet loss from out-of-order packets and initiates retransmission. The effect on throughput may be greater in high-speed networks because of the potential for a higher rate of reordering, which would necessitate more frequent retransmissions [94].

Increased latency: Rearranging packets might cause extra latency since TCP must wait for the packets that are out of order to be rearranged before sending them to the application layer [95]-[97]. This may be especially noticeable in high-speed networks, where low-latency communication is frequently a crucial need [98], [99].

Duplicate acknowledgments: TCP might get two acknowledgments for the same data segment if there is packet duplication [100]. This may result in pointless retransmissions and affect how well TCP's congestion control algorithms work [101]. Because the time window for receiving duplicate acknowledgments may be shorter on high-speed networks, more aggressive retransmission behavior may result.

Out-of-order delivery: Data is delivered via TCP in the same order as it was sent [102]. This order may be upset by packet reordering, which would result in ineffective data transfer to the application layer. Applications that depend on data sequencing, such multimedia streaming and real-time communication, may find this especially troublesome.

Reduced TCP performance: For optimal performance, TCP depends on precise acknowledgment and sequencing algorithms [103]. Subpar performance can result from these techniques being confused by packet reordering and duplication. Reordering and duplication events may occur more frequently in high-speed networks due to their larger data rates, which can have an increasing effect on TCP performance.

Increased complexity for TCP implementations: Complicating TCP implementations are handling out-of-order packets and controlling packet duplication [104]. It becomes imperative to build strong and efficient techniques for handling reordering in high-speed networks, because the frequency of such events may be higher.

5. Some solutions to TCP security issues

TCP security issues can be addressed through various means. One approach involves implementing encryption mechanisms such as Transport Layer Security (TLS) to secure data in transit, mitigating eavesdropping and tampering risks. Additionally, employing firewalls and intrusion detection/prevention systems helps to filter and monitor network traffic, safeguarding against unauthorized access and malicious activities. Regular security audits and updates to patch vulnerabilities in TCP/IP implementations are crucial, along with enforcing strong authentication mechanisms like two-factor authentication (2FA) to prevent unauthorized access to network resources [105]-[109]. Moreover, adopting best practices such as segmenting networks, limiting access privileges, and maintaining robust access controls bolster overall TCP security posture, reducing the likelihood of breaches and data compromise.

5.1. Implementation of security measures to enhance the robustness of TCP connections

Enhancing the resilience of TCP connections through the implementation of multiple security measures is necessary to mitigate TCP sequence prediction attacks in high-speed networks. Some suggested remedies that can assist in lowering the dangers resulting from security vulnerabilities include the following:

Randomization of initial sequence numbers: To generate initial sequence numbers for TCP connections, use robust randomization algorithms [110], [111]. As a result, attackers will find it far more difficult to correctly guess the sequence numbers and carry out sequence prediction attacks.

TCP timestamps: To further increase the sequence numbers' complexity, use TCP timestamps. Since attackers must take into account the timestamp value, timestamps can make it more challenging for them to predict the sequence number with accuracy [112], [113].

Encryption and authentication: Employ encryption (e.g., TLS/SSL) to secure the communication between parties [114], [115]. Encrypted connections make it more challenging for attackers to manipulate or inject malicious data into the communication stream. Implement strong authentication mechanisms to verify the identity of communicating parties, reducing the risk of session hijacking.

Intrusion Detection and Prevention Systems (IDPS): Install cutting-edge intrusion detection and prevention systems that can manage a lot of traffic. These systems have real-time capabilities to monitor network behavior, identify anomalies, and react to possible TCP sequence prediction assaults [116], [117].

Rate limiting and traffic shaping: Control network traffic flow by implementing traffic shaping and rate limiting methods [118]. Attackers find it more difficult to overwhelm the network and carry out denial-of-service [119] assaults when the pace of incoming packets is regulated.

Stateful firewalls: Use Stateful firewalls that can inspect and track the state of TCP connections [120]. Stateful inspection allows firewalls to identify and block malicious packets by analyzing the context of the communication, including sequence numbers.

Regular software updates and patching: Update any network hardware, such as firewalls, switches, and routers, with the most recent security updates [121]. Updates on a regular basis aid in mitigating known vulnerabilities that sequence prediction attacks might exploit.

Network monitoring and anomaly detection: Use network monitoring technologies that can identify odd actions and trends [122]. Systems for detecting anomalies in network traffic can spot abnormalities and send out automated answers or notifications [123].

5.2. Enhanced Congestion Control Algorithms

In order to curb the increased vulnerability to Distributed Denial of Service (DDoS) attacks in TCP when deployed in high-speed networks, there is a need to use a multifaceted approach that involves both network architecture and security mechanisms.

Traffic filtering and rate limiting: At the network perimeter, put in place traffic filtering measures to detect and stop malicious traffic linked to DDoS assaults [124], [125]. By limiting the pace at which incoming traffic is handled, rate limitation can be used to avoid overusing resources.

Intrusion Prevention Systems (IPS) and firewalls: Install firewalls and intrusion prevention systems that examine incoming communications for irregularities and recognized attack patterns [126], [127]. By recognizing and blocking malicious packets, these systems can lessen the effect of DDoS attacks.

Anycast routing: Implement Anycast routing to distribute incoming traffic across multiple servers or data centers [128], [129]. This distributes the load and makes it more challenging for attackers to concentrate their efforts on a single point, enhancing resilience against DDoS attacks.

Content Delivery Networks (CDNs): To cache and distribute content among geographically dispersed servers, use content delivery networks [130]. A sizable amount of DDoS traffic [131] might be absorbed by CDNs, lessening the toll on the main servers and enhancing service availability.

Cloud-Based DDoS protection services: Leverage cloud-based DDoS protection services that specialize in mitigating large-scale attacks [132], [133]. These services often have the capacity and expertise to absorb and filter out malicious traffic, keeping the network and services accessible.

Border Gateway Protocol Flow spec (*BGP Flow spec*)*:* To dynamically distribute flow specification rules around the network, implement BGP Flow Spec [134], [135]. This makes it possible to control network flows more precisely and to identify and stop DDoS attack patterns instantly.

Rate-based traffic policing: Use rate-based traffic policing techniques to set thresholds for what constitutes an acceptable amount of incoming traffic [136]. By doing this, the network is kept from becoming overloaded with requests all at once.

Behavioral analysis and machine learning: Employ behavioral analysis and machine learning algorithms [137] to detect abnormal patterns in network traffic. These systems can adapt to evolving DDoS attack strategies by learning from the network's historical patterns [138].

Incident response and communication plans: Develop robust incident response plans to facilitate a coordinated and timely response to DDoS attacks [139]. Clear communication channels and predefined response strategies ensure a more effective defense against ongoing attacks.

Collaboration with ISPs and security communities: Collaborate with Internet Service Providers (ISPs) and participate in security communities to share threat intelligence [140]. Timely information exchange can help in identifying and mitigating DDoS attacks at an early stage.

5.3. DDoS Mitigation Strategies

DDoS mitigation strategies encompass a multifaceted approach aimed at minimizing the impact of such attacks on network and service availability. Firstly, network-level mitigation involves filtering and traffic scrubbing to differentiate between legitimate and malicious traffic, often utilizing specialized hardware or cloud-based services. Additionally, application-layer protection involves configuring web servers and applications to withstand high volumes of requests and implementing rate limiting and authentication mechanisms [141]-[143]. Employing redundancy and failover systems ensures service continuity, while threat intelligence sharing and collaborative defense mechanisms allow for timely response and adaptation to evolving attack vectors. Continuous monitoring and analysis of traffic patterns are integral to detecting and mitigating DDoS attacks swiftly, thereby safeguarding the stability and accessibility of online services.

Traffic scrubbing services: Make use of traffic scrubbing services offered by knowledgeable DDoS mitigation companies [144]. These services filter out DDoS attack traffic before it reaches the target network, analyze incoming traffic, and spot harmful patterns.

Rate limiting and traffic shaping: Control the rate of incoming traffic by implementing traffic shaping and rate limitation methods [145]. By establishing acceptable traffic thresholds, the network can stop excessive requests and lessen the damage caused by DDoS attacks.

Behavioral analysis: Use behavioral analysis to detect deviations from normal traffic patterns [146]. Behavioral analysis systems can identify anomalies in real-time and trigger automatic responses or alerts to mitigate potential DDoS attacks.

Anomaly detection systems: Distribute incoming traffic across multiple servers or data centers using techniques such as Anycast routing [147]. This helps distribute the load and makes it more challenging for attackers to concentrate their efforts on a single point.

Application layer protection: Implement application layer protection mechanisms to filter out malicious traffic at the application level [148]. This can include the use of web application firewalls (WAFs) and application-layer DDoS protection solutions.

Rate-based filtering: Use rate-based filtering to identify and block traffic based on predefined rate thresholds. This can help in preventing the network from being overwhelmed by a sudden surge in requests [150], characteristic of DDoS attacks.

TCP connection management: Optimize TCP connection management by setting reasonable timeouts for idle connections [151]. This helps in freeing up resources and preventing attackers from tying up resources with idle connections as part of a DDoS attack.

Automated incident response: Implement automated incident response mechanisms that can detect and respond to DDoS attacks in real-time [152]. Automation allows for swift responses, reducing the impact of attacks and minimizing downtime.

Collaboration with ISPs: Collaborate with Internet Service Providers (ISPs) to implement upstream filtering and traffic diversion during DDoS attacks [153]. ISPs can play a crucial role in mitigating attacks before they reach the target network.

Cloud-based DDoS protection: Leverage cloud-based DDoS protection services that can absorb and filter out malicious traffic before it reaches the on-premises infrastructure [154]. Cloud services often have the capacity to handle large-scale DDoS attacks effectively.

5.4. Packet Reordering and Duplication Strategies

Packet reordering and duplication strategies are essential components of network optimization and fault tolerance mechanisms [155]. In scenarios where packets encounter varying network paths or experience delays, reordering strategies prioritize packet delivery based on sequence numbers or timestamps [156], ensuring that data is reconstructed correctly at the receiving end. This prevents performance degradation and minimizes the risk of data loss or corruption. Moreover, duplication strategies involve replicating packets at various points along the network route, enhancing fault tolerance by providing redundant data streams. These strategies are particularly crucial in real-time applications, such as multimedia streaming or online gaming, where maintaining packet order and minimizing latency are paramount for a seamless user experience.

Selective Acknowledgment (SACK): The TCP protocol has an addition called Selective Acknowledgment (SACK) that enables the recipient to notify the sender of non-contiguous data blocks that have been successfully received [157]. This minimizes the number of pointless retransmissions by allowing the sender to retransmit just the segments that are missing or out of order.

Forward Error Correction (FEC): FEC transmits redundant data in addition to the original data so that errors can be corrected by the receiver without requiring retransmission [158]. FEC can be used to lessen the effects of reordering and packet loss, particularly on high-speed networks where retransmissions could cause delays.

Packet reordering buffer: rearranging out-of-sequence packets at the recipient using a buffer before sending them to the higher tiers [159]. By keeping packets until they can be transmitted in the right order, this buffer can assist lessen the effects of reordering.

Increasing TCP window size: More outstanding packets can be present in the network at any given time with a bigger TCP window size, which might be advantageous when reordering happens [160]. Depending on the features of the particular network, adjusting the TCP window size could need fine-tuning.

Timestamps and Round-Trip Time (RTT) measurements: RTT measurements and TCP timestamps can be utilized to detect and control reordering [161]. Both the sender and the recipient can reorder packets in accordance with network delays by adding timestamp information.

Path Maximum Transmission Unit (PMTU) discovery: confirming that the TCP Maximum Segment Size (MSS) is configured correctly and figuring out the best path based on packet size using PMTU discovery [162], [163]. Reordering may result from fragmentation and reassembly at routers; minimizing these problems can be achieved by raising the MTU.

Explicit Congestion Notification (ECN): Routers can alert endpoints of approaching congestion using ECN without losing packets [164]. By doing this, needless retransmissions brought on by packet loss resulting from congestion may be avoided.

Use of middleboxes and load balancers: Certain middleboxes, such load balancers, have the potential to cause reordering [165]. It is crucial to make sure that these devices are set up correctly and do not result in excessive reordering.

6. The performance issues of TCP when deployed in high-speed networks

For reliability, many high-speed networks rely on the TCP/IP protocol, which is implemented in software and thus buffer size sensitive. To ensure almost 100% link optimization, TCP requires a buffer size of bandwidth delay product in switches/routers. However, a buffer this size will complicate hardware design, increase power consumption, and cause jitter and queueing delays.

6.1. Bandwidth Delay Product

At jitter speeds, the performance of TCP's congestion control algorithms may degrade sharply. The fault lies mainly in the way TCP dynamically adapts its congestion window, both in the absence of packet loss and when the losses occur [166]. As shown in Figure 4, in the congestion avoidance phase, a TCP sender increases the congestion window roughly by only one segment per round-trip time (RTT). When the sender detects packet loss, the congestion window is cut by at least a half. Whenever the BDP of the end-to-end path is high, the combination of these two policies for updating the congestion window often result in poor performance [167]. When a packet is lost, it may take many RTT cycles before the sender attains a large window.

Jacobson's classical rule-of-thumb states that the size B of a buffer in a bottleneck router should be: $B = C*RTT$, where C=The rate of the egress bottleneck link, and RTT= The average RTT experienced by connections that use that link. The basis of this rule expresses the minimum amount of buffering needed to avoid link underutilization, assuming there is a single long-lived flow (or a set of fully synchronized flows) using the link. In high BDP links, such rule often yields huge impractical values of B.

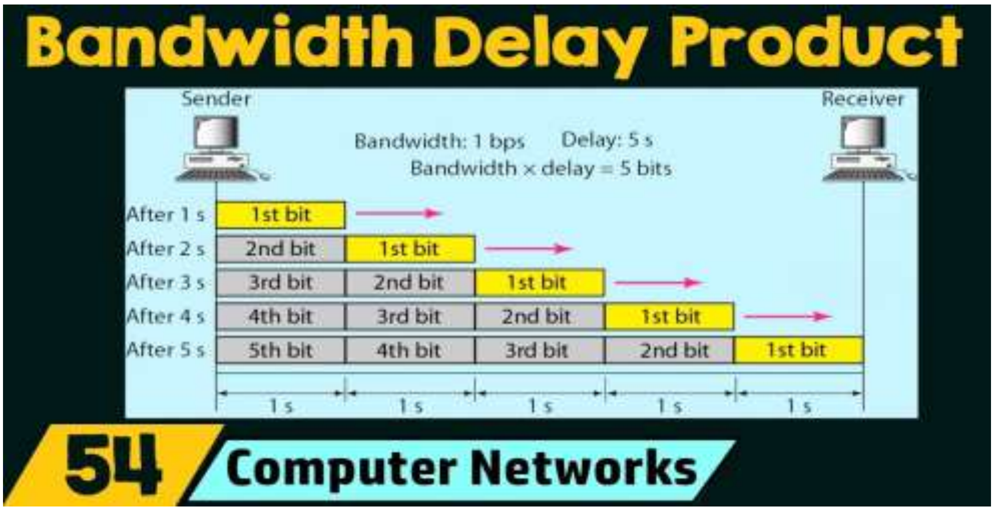


Figure 4 Bandwidth Delay Product

Other ways in which BDP influences the performance in TCP when deployed in high speed networks include:

Optimal throughput and window size: More data can potentially be transmitted before being acknowledged, according to a higher BDP [168]. The amount of unacknowledged data, or TCP window size, must be scaled appropriately in order to make the most of the available bandwidth. Suboptimal throughput [169] may result from the network not being fully utilized if the window size is too small in comparison to the BDP.

TCP slow start and initial congestion window: The rate at which a connection can increase its throughput is largely dependent on the TCP Slow Start phase [170]. TCP may take longer to reach an ideal window size during the Slow Start phase in high-speed networks with a big BDP. Particularly for connections that are temporary, this delay may have an effect on the initial data transfer rate and lead to less than ideal performance.

Round-trip time sensitivity: Performance of TCP is very dependent on round-trip time. Even a little RTT in high-speed networks can lead to a big BDP [171]. TCP finds it more difficult to efficiently modify its window size and congestion control parameters the longer the round-trip time.

Efficient bandwidth utilization: TCP seeks to use available bandwidth as effectively as possible without creating congestion [172]. High throughput in high-speed networks is dependent on TCP's capacity to maintain an acceptable quantity of data in transit, and the BDP directly affects the amount of "in-flight" data.

Retransmission Timeout (RTO) calculation: For the purpose of identifying and recuperating from packet loss, the RTO is computed using the BDP. Because of the enormous BDP in high-speed networks, calculating RTO might be difficult [173]. An RTO set too high could cause a delay in recovering from packet loss, while an RTO set too low could result in pointless retransmissions.

Congestion control efficiency: High BDP may cause traditional TCP congestion control strategies to scale poorly. Underutilization of the network or, on the other hand, significant congestion might result from ineffective congestion control [174]. Congestion control mechanism optimization is a prerequisite for attaining effective performance in high-speed networks [175].

Buffer sizing and queuing delays: The necessary buffer size at switches and routers along the network path is determined by the BDP. Reduced performance and missed packets could result from too tiny of buffers [176]. However, excessive buffering can have a detrimental effect on TCP performance by increasing queuing delays and causing buffer bloat.

6.2. Packet Loss and Retransmissions

Packet loss in high-speed networks can be caused by a number of things, including noise, congestion, and network failures [177]. Because TCP depends on retransmitting dropped packets, throughput can be negatively impacted in high-speed situations due to the time it takes to identify and recover from losses. TCP performance can be significantly impacted by packet loss and retransmission [178], particularly in high-speed networks. TCP is a dependable, connection-oriented protocol developed to guarantee data transmission reliability and integrity.

Round-Trip Time (RTT) impact: The Round-Trip Time (RTT) may still be minimal in high-speed networks, but it becomes increasingly important. The round trip time (RTT) of a packet is the time spent traveling from sender to recipient and back [179]. TCP starts a retransmission, which increases the RTT, if it detects a packet loss and believes it is the result of congestion. Even slight improvements in RTT can have an impact on the connection's throughput on high-speed networks.

Congestion window size: Depending on the state of the network, TCP's congestion control mechanism modifies the size of its congestion window. TCP lowers its congestion window when a packet is dropped, which results in a brief drop in the quantity of data that can be transferred [180]. Rapid changes to the congestion window in high-speed networks could result in less-than-ideal throughput even though the network can handle more data.

Bandwidth utilization: Although high-speed networks are built to accommodate a lot of data, retransmission and packet loss can cause wasteful use of the available capacity [181], [182]. TCP starts retransmissions when a packet is dropped, which may cause the network to use less bandwidth as it waits for the retransmitted packets to be acknowledged.

Performance oscillations: TCP's congestion control algorithms may have trouble telling the difference between temporary problems like packet reordering or random packet loss and true network congestion in high-speed networks [183[. Due to this, the congestion window may be needlessly reduced, which could result in performance oscillations where the throughput fluctuates between high and low values.

Buffering and queuing issues: Increased buffer sizes are used in high-speed networks to accommodate spikes in traffic. On the other hand, excessive overloading of these buffers may lead to longer wait times and even packet loss [184]. Even if the network could handle the data, TCP perceives this packet loss as an indication of congestion and reacts by decreasing its transmitting rate.

Long Fat Networks (LFNs): The TCP performance may be especially impacted on networks with large capacity and long round-trip times (LFNs) [185]. Significant RTT in conjunction with high-speed networks can result in wasteful utilization of available bandwidth, particularly in the event of packet loss.

6.3. Window Size Limitations

To regulate how much data is in the network that is not acknowledged, TCP employs a sliding window method [186]. The default window size on high-speed networks can become restricting, and raising the window size necessitates careful tuning to strike a balance between dependability and efficiency. Because window size is a critical parameter in TCP that controls the amount of unacknowledged data that can be in transit between the sender and receiver, its constraints can have a substantial effect on the protocol's performance [187].

Bandwidth-Delay Product (BDP): The Bandwidth-Delay Product (BDP), which is the result of multiplying the available bandwidth by the round-trip time (RTT), is directly proportional to the window size [188[. A tiny window size may not completely utilize the available bandwidth in high-speed networks [189] with large bandwidth, which could lead to underutilization of the network capacity.

Throughput limitations: The TCP throughput is restricted by the window size. A high-speed network may have limited throughput and a connection that falls short of its promise if the window size is not set adequately [191]. Inefficient data transport might result from a decreased window size, particularly when using high-latency or long-distance connections.

RTT impact: Even though the RTT may not increase much on high-speed networks, it is still a crucial component of TCP performance. To ensure that there is enough data in transit without having to wait for acknowledgments, the window size needs to be sufficiently large [191]. A TCP connection may not completely utilize the available bandwidth if the window size is too tiny, which would result in less than ideal performance.

TCP slow start and congestion avoidance: TCP gradually increases the window size until it reaches a number that makes the best use of the network capacity through a method known as slow start. Slow starts in high-speed networks can cause a delayed ramp-up of the window size, which would impair throughput in the connection's early stages [192]. In order to properly manage high-speed links, congestion avoidance algorithms might also need to be adjusted.

Buffering challenges: Large buffers are a common feature of high-speed networks to manage the intermittent nature of traffic. TCP may not completely utilize these buffers, though, if the window size is too short, which would result in an inefficient use of the resources that are available [193]. Issues with buffering can lead to poor TCP performance and underuse of the network.

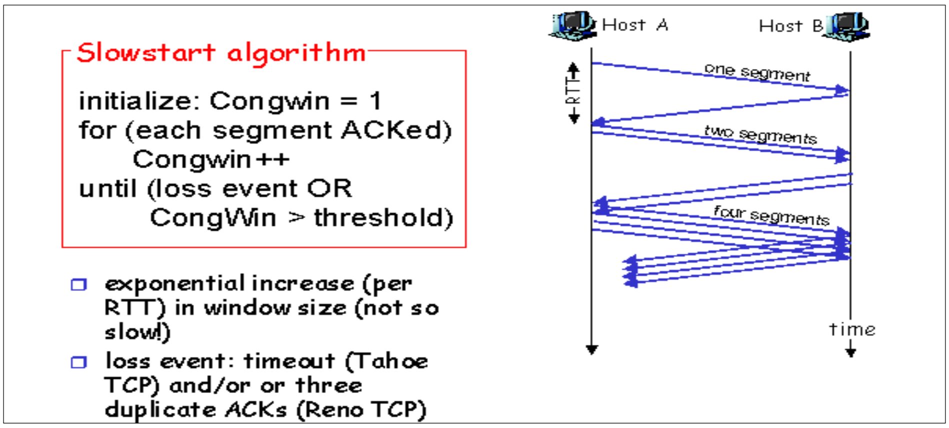
Packet loss and retransmission: High-speed networks that have inadequate window sizes may have more packet loss and retransmission [194]. Retransmissions brought on by packet loss may increase in frequency if the window size is too short, resulting in wasteful network use and decreased throughput overall.

7. TCP Slow Start

TCP Slow Start is an algorithm that gradually increases the congestion window size when a connection is initiated or after a timeout, as shown in Figure 5. In high-speed networks, the Slow Start phase may take longer, leading to suboptimal performance during the initial stages of data transfer. TCP Slow Start is a mechanism used by TCP to control the rate at which a sender increases its congestion window size during the initial phase of a connection [195]. While Slow Start is essential for network stability and fairness, its impact on performance can be particularly pronounced in high-speed networks.

Gradual window size increase: In the Slow Start phase, TCP starts with a small congestion window and gradually increases it exponentially. This initial conservative approach helps prevent network congestion [196], [197]. However, in high-speed networks, the slow and exponential increase may cause the connection to take longer to reach its optimal throughput.

Underutilization of bandwidth: High-speed networks are designed to handle large amounts of data. The slow and cautious increase in the congestion window during Slow Start might result in underutilization of the available bandwidth [198], especially during the initial phase of the connection. The network might be capable of transmitting data at a faster rate, but TCP Slow Start delays the process.



Slowstart algorithm

```
initialize: Congwin = 1
for (each segment ACKed)
      Congwin++
until (loss event OR
        CongWin > threshold)
```

- □ exponential increase (per RTT) in window size (not so slow!)
- □ loss event: timeout (Tahoe TCP) and/or or three duplicate ACKs (Reno TCP)

Figure 5 TCP Slow Start

Impact on short-lived connections: For short-lived connections, the Slow Start phase becomes a more significant portion of the overall connection time in high-speed networks [199]. This can lead to suboptimal throughput for short-lived connections, where the connection may be established and terminated before reaching its potential throughput.

Round-Trip Time (RTT) sensitivity: Slow Start's duration is influenced by the Round-Trip Time (RTT) of the network. In high-speed networks, the RTT might be relatively small, but the slow and exponential growth of the congestion window can still affect performance [200]. Adjusting Slow Start parameters to be more adaptive to high-speed networks may be necessary.

Impact on bursty traffic: Slow Start might not be well-suited for bursty traffic patterns often seen in high-speed networks [201]. Bursty traffic could lead to frequent retriggering of Slow Start, resulting in a less efficient use of the available bandwidth.

Congestion window limitations: The maximum size of the congestion window reached during Slow Start can be limited by factors such as the Maximum Segment Size (MSS) and other network parameters [202]. In high-speed networks, if this maximum size is not adjusted appropriately, it might prevent TCP from fully utilizing the available bandwidth.

Congestion avoidance transition: After the Slow Start phase, TCP transitions to the Congestion Avoidance phase, where the congestion window grows linearly [203]. However, the transition from Slow Start to Congestion Avoidance might be delayed in high-speed networks, impacting the overall performance [204] during the connection.

8. Congestion Control Challenges

Traditional TCP congestion control mechanisms may not be well-suited for high-speed networks. In particular, the conservative nature of congestion control algorithms can result in underutilization of available bandwidth. Figure 6 gives an overview of the congestion control process. Congestion control challenges can significantly impact the performance of TCP when deployed in high-speed networks [205]. Congestion control is a crucial aspect of TCP's operation as it ensures fair and efficient use of network resources. However, in high-speed networks, certain challenges can arise that affect TCP's ability to manage congestion effectively [206].

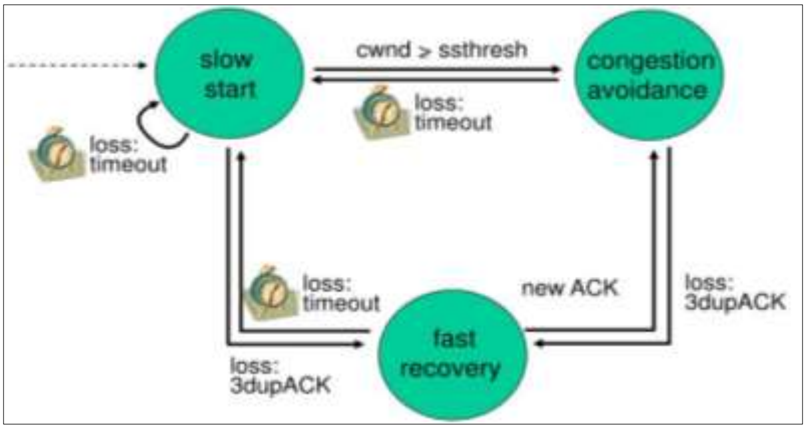


Figure 6 Congestion Control overview

Round-Trip Time (RTT) sensitivity: TCP relies on observing Round-Trip Times (RTT) to make decisions about the network's congestion state. In high-speed networks, the RTT might still be relatively small, but the combination of high bandwidth and small RTT can make TCP more sensitive to small variations [207]. This sensitivity can lead to oscillations in the congestion window and suboptimal performance.

Slow start challenges: Slow Start is a mechanism in TCP where the sender initially starts with a small congestion window and gradually increases it exponentially [208]. In high-speed networks, Slow Start may lead to underutilization of the available bandwidth as it takes time for the congestion window to grow to a level that fully exploits the network capacity.

Queue management: High-speed networks often have larger buffers to handle bursts of traffic. However, if these buffers become full, it can lead to increased queuing delays and the possibility of packet loss [209]. TCP interprets packet loss as a sign of congestion and responds by reducing its sending rate, impacting overall throughput.

TCP incast: The term "TCP Incast" describes a networking phenomenon that happens in data center settings, especially when several servers are requesting data from one server at once. Congestion and decreased network performance may result from this circumstance [210]. It is basically the synchronization of TCP acknowledgements. In scenarios where multiple senders are trying to transmit data to a common receiver simultaneously, a phenomenon known as TCP Incast can occur. This situation can lead to congestion at the receiver [211], causing a reduction in throughput for all senders. High-speed networks can exacerbate the effects of TCP Incast due to the rapid arrival of packets.

Short flows and fairness: Short-lived flows in high-speed networks may experience difficulties in achieving fairness with long-lived flows [212]. Congestion control mechanisms in TCP may not be optimized for short flows, leading to less efficient utilization of the available bandwidth for these types of connections.

Large windows and bufferbloat: High-speed networks may allow for large congestion windows, leading to the potential for bufferbloat [213]. Bufferbloat occurs when large buffers in networking devices are filled, causing increased latency. TCP may not always respond optimally to bufferbloat, resulting in degraded performance.

Congestion window limitations: The Maximum Segment Size (MSS) and other factors can limit the maximum size of the congestion window in TCP [214]. In high-speed networks, if these limitations are not appropriately adjusted, the congestion window might not scale to fully utilize the available bandwidth.

9. Queuing Delays

Queues at routers and switches can introduce delays, especially in the presence of bursty traffic or congestion. High-speed networks may experience these queuing delays, impacting overall latency and throughput [215]. As shown in Figure 7, queuing delays can significantly influence the performance of TCP in high-speed networks. Figure 8 shows the calculations of the queuing delays, which occurs when packets experience delays in network queues, either at routers or switches, before being transmitted to their destination. In high-speed networks, the interaction between TCP and queueing delays can lead to several performance issues:

Increased Round-Trip Time (RTT): Queuing delays contribute to the overall Round-Trip Time (RTT) experienced by TCP connections. In high-speed networks, where the RTT might be relatively small, additional delays due to queuing can become a more significant portion of the total RTT [216]. This can impact TCP's ability to accurately estimate the network conditions and adjust its congestion window size.

Bufferbloat: High-speed networks often use large buffers to handle bursts of traffic efficiently. However, if these buffers become too large, it can result in a phenomenon known as bufferbloat [217]. Bufferbloat occurs when excessively large buffers introduce additional queuing delays, leading to increased latency and jitter. TCP may interpret these delays as signs of congestion, triggering unnecessary reductions in the congestion window.

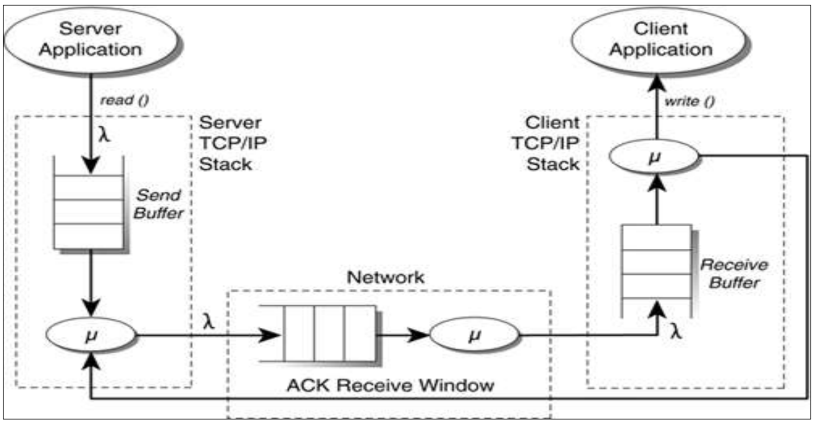


Figure 7 Queuing delays in TCP

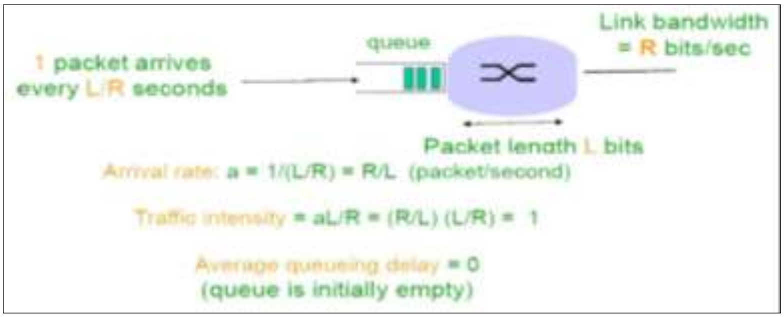

1 packet arrives every L/R seconds
queue
Link bandwidth = R bits/sec
Packet lenath L bits
Arrival rate: a = 1/(L/R) = R/L (packet/second)
Traffic intensity = aL/R = (R/L) (L/R) = 1
Average queueing delay = 0
(queue is initially empty)

Figure 8 Calculating queuing delays

TCP inefficiency: Queuing delays can cause inefficiencies in TCP's congestion control mechanisms. Long queues can lead to delays in receiving acknowledgment signals, preventing TCP from accurately gauging the available bandwidth and causing suboptimal throughput [218]. TCP may not fully utilize the available capacity due to conservative window adjustments based on perceived congestion.

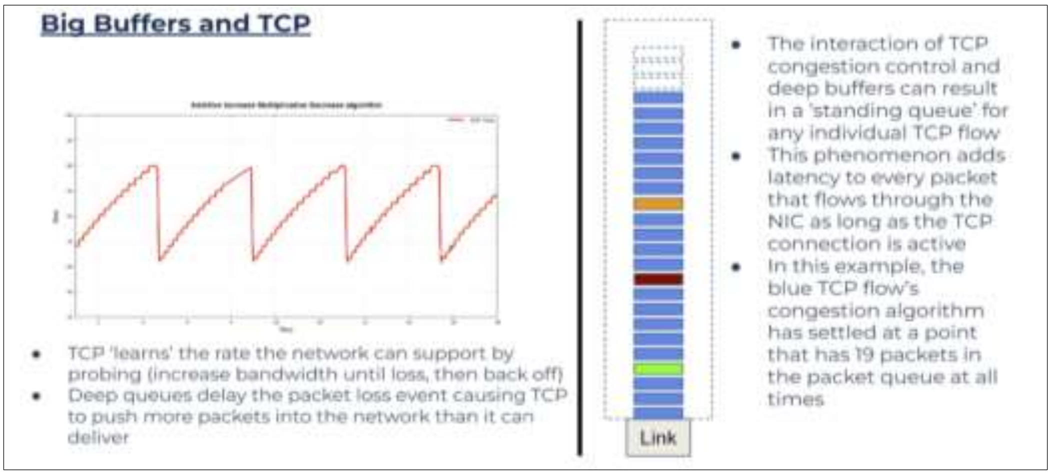
Head-of-Line blocking: Queuing delays can result in head-of-line blocking, where a delayed packet holds up the transmission of subsequent packets [219]. This can impact the efficiency of TCP, particularly in scenarios with multiple flows. Head-of-line blocking can prevent timely delivery of packets [220], affecting overall throughput and responsiveness.

Fairness issues: Queueing delays can introduce fairness issues, especially when there are competing flows sharing a bottleneck link [221]. Flows experiencing longer queueing delays may receive less bandwidth compared to those with shorter delays, leading to an unfair distribution of network resources. This can affect the fairness and equality of TCP connections in high-speed networks.

Tail drop and packet loss: In extreme cases, when the queue sizes are limited, high-speed networks may experience tail drop, where incoming packets are dropped if the queue is full [222]. This can lead to packet loss, triggering TCP's congestion control mechanisms and causing reduced throughput. Tail drop can occur if the queue is not sized appropriately for the network conditions.

10. Bufferbloat

Excessive buffering in network devices, known as bufferbloat, can occur in high-speed networks. This can lead to increased latency and jitter, affecting TCP's ability to adapt to changing network conditions [223]. As shown in Figure 9, bufferbloat is a phenomenon that can significantly influence the performance of TCP (Transmission Control Protocol) in high-speed networks. Bufferbloat occurs when excessively large buffers are used in network devices such as routers and switches, leading to increased latency, jitter, and suboptimal performance.



Big Buffers and TCP

- TCP 'learns' the rate the network can support by probing (increase bandwidth until loss, then back off)
- Deep queues delay the packet loss event causing TCP to push more packets into the network than it can deliver

- The interaction of TCP congestion control and deep buffers can result in a 'standing queue' for any individual TCP flow
- This phenomenon adds latency to every packet that flows through the NIC as long as the TCP connection is active
- In this example, the blue TCP flow's congestion algorithm has settled at a point that has 19 packets in the packet queue at all times

Figure 9 Bufferbloats in TCP

Increased latency: Excessive buffering introduces additional latency as packets spend more time waiting in queues before being transmitted [224]-[226]. In high-speed networks, where low latency is a key expectation, bufferbloat can result in longer Round-Trip Times (RTT). Increased latency affects the responsiveness of TCP connections, impacting real-time applications such as video conferencing and online gaming.

Impact on TCP congestion control: TCP uses congestion control mechanisms to adapt its sending rate based on network conditions. Bufferbloat can lead to delayed feedback, causing TCP to interpret increased latency as a sign of congestion [227]. In response, TCP may unnecessarily reduce its sending rate, leading to underutilization of available bandwidth and reduced throughput.

Jitter and variability: Bufferbloat introduces jitter, which refers to variations in packet arrival times. Jitter can negatively impact the performance of real-time applications that rely on consistent and predictable data delivery [228]. High-speed networks with bufferbloat may experience increased variability in packet arrival times, affecting the quality of service for applications sensitive to delay variations.

Inefficient use of bandwidth: Large buffers can lead to bursts of packet transmissions followed by periods of inactivity. This bursty behavior can result in inefficient use of available bandwidth [229]. TCP may not fully exploit the network's capacity during periods of low congestion, leading to suboptimal performance and lower throughput than what the network could otherwise support.

Unnecessary retransmissions: Bufferbloat can contribute to packet loss when buffers become full and overflow [230]. TCP interprets packet loss as a sign of congestion, triggering retransmissions. However, in the presence of bufferbloat, the loss may be due to temporary overflow rather than persistent congestion. Unnecessary retransmissions can further exacerbate the inefficiency of TCP in high-speed networks.

Fairness issues: Bufferbloat can introduce fairness issues, particularly when multiple flows sheer a bottleneck link. Flows experiencing varying degrees of bufferbloat may receive different levels of service, leading to an unfair distribution of network resources [231]. This can affect the equality of TCP connections in high-speed environments.

11. Inefficient Window Scaling

In some cases, inefficient window scaling or misconfigured Maximum Segment Size (MSS) settings can hinder TCP performance in high-speed environments. Inefficient window scaling can have a significant impact on the performance [232] of TCP when deployed in high-speed networks. Window scaling is a mechanism used by TCP to extend the range of its window size, allowing it to better utilize the available bandwidth, especially in networks with high speeds [233]. However, if window scaling is not implemented or configured efficiently, it can lead to suboptimal TCP performance in high-speed environments [234].

Limited throughput: In high-speed networks, the default window size of TCP may become a bottleneck, limiting the amount of data in transit at any given time [235]. If window scaling is not employed or is inefficiently configured, the congestion window may not grow to a size that fully utilizes the available bandwidth, resulting in suboptimal throughput.

Underutilization of bandwidth: Window scaling allows TCP to use larger window sizes to take advantage of the higher bandwidth available in high-speed networks. Inefficient window scaling may lead to underutilization of the network's capacity, as the congestion window may not scale appropriately [236]. This can result in the network not reaching its full potential in terms of data transfer rates.

Increased Round-Trip Time (RTT) sensitivity: Inefficient window scaling can make TCP more sensitive to Round-Trip Time (RTT) variations. In high-speed networks, where the RTT might still be relatively small, inefficient window scaling may lead to suboptimal performance, as TCP may not adjust its window size adequately to accommodate the network conditions [237], [238].

Delayed congestion window growth: Window scaling is essential for the efficient growth of the congestion window during the Slow Start phase of TCP [239]. Inefficient window scaling may cause the congestion window to grow more slowly than necessary, delaying the sender's ability to fully exploit the available bandwidth, particularly during the initial phase of the connection.

Inefficient handling of packet loss: In the presence of packet loss, TCP needs to adapt its congestion window to mitigate the impact of network congestion. Inefficient window scaling may lead to a less responsive adjustment, causing TCP to underutilize the available bandwidth after encountering packet loss, resulting in degraded performance [240].

Failure to accommodate bursty traffic: High-speed networks often experience bursts of traffic. Inefficient window scaling may prevent TCP from efficiently handling bursty traffic patterns, limiting its ability to send a significant amount of data during periods of congestion-free network conditions [241].

12. Head-of-Line Blocking

Head-of-line blocking occurs when a single lost or delayed packet prevents the delivery of subsequent packets in the same TCP connection. In high-speed networks, the impact of head-of-line blocking can be more pronounced [242]. Head-of-Line (HoL) blocking can significantly influence the performance of TCP in high-speed networks. HoL blocking occurs when a delayed or lost packet in a TCP flow holds up the delivery of subsequent packets, creating inefficiencies in the data transfer process. In high-speed networks, where the capacity for rapid data transmission is substantial, HoL blocking can have several implications:

Throughput limitations: HoL blocking can limit the overall throughput of a TCP connection in high-speed networks. If a packet is delayed or lost, subsequent packets are held back until the missing or delayed packet is retransmitted or acknowledged [243]. This can lead to underutilization of the available bandwidth and result in lower throughput [244] than the network is capable of supporting.

Impact on TCP window size dynamics: TCP relies on efficient acknowledgment and congestion control mechanisms to dynamically adjust its congestion window size [245]. HoL blocking disrupts the smooth flow of acknowledgments, preventing TCP from accurately assessing the network conditions. This can lead to suboptimal window size adjustments, affecting TCP's ability to fully exploit the available bandwidth in high-speed networks.

Delayed flow completion: HoL blocking can delay the completion of a TCP flow. In high-speed networks where rapid data transmission is expected, delays in flow completion may result in decreased efficiency and responsiveness of applications relying on timely data delivery [246].

Real-time application performance: Real-time applications, such as video streaming or VoIP, are particularly sensitive to delays [247]. HoL blocking can introduce additional latency, negatively impacting the performance of these applications in high-speed networks. Consistent data delivery is crucial for maintaining the quality of service for time-sensitive applications.

Inefficient resource utilization: HoL blocking may lead to inefficient utilization of network resources, as the available bandwidth is not fully leveraged during periods of congestion-free transmission [248]. This inefficiency becomes more pronounced in high-speed networks, where the network can support rapid data transfer.

Fairness issues: In scenarios with multiple TCP flows sharing a bottleneck link, HoL blocking can create fairness issues [249]. Some flows may experience delays due to blocked packets, leading to an uneven distribution of available bandwidth and affecting the fairness of the networks.

13. Proposed solutions

Deploying TCP in high-speed networks presents significant challenges related to both security and performance. At high speeds, TCP's congestion control mechanisms may struggle to keep pace with rapidly changing network conditions, leading to inefficient bandwidth utilization [250] and potential performance degradation. Additionally, the increased volume of traffic in high-speed environments can exacerbate security vulnerabilities, such as amplification attacks and resource exhaustion, making networks more susceptible to DDoS (Distributed Denial of Service) attacks. Moreover, the reliance on packet headers for flow control and error detection becomes more pronounced, increasing the risk of header manipulation and packet spoofing, which can compromise data integrity and confidentiality. Consequently, deploying TCP in high-speed networks necessitates robust security measures and optimization techniques to mitigate these risks while maintaining optimal performance and reliability. The following are some of the probable solutions to these issues.

13.1. Bandwidth Delay Product issue

The TCP window size can be increased above the standard 64 KB limit with the use of this technology. By scaling the TCP window, it makes better use of the network by supporting larger bandwidth-delay products [251]. In addition, we can use TCP Acknowledgment Selective (SACK). Basically, using the TCP SACK extension, a recipient can acknowledge multiple non-contiguous blocks of data. SACK enhances the ability of high-speed networks to recover from multiple packet losses, hence increasing overall throughput [252]. Another option will be to use

TCP Cubic and BBR Congestion Control Algorithms. Modern congestion management methods like TCP Cubic and BBR (Bottleneck Bandwidth and Round-trip propagation time) are made to optimize TCP performance on high-speed, high-latency networks [253]. These algorithms improve throughput and lower latency by dynamically modifying their behavior in response to network conditions.

13.2. Packet Losses and Retransmission

The possible solutions here include Forward Error Correction (FEC), Selective Acknowledgment (SACK) as well as TCP Fast Retransmit and Recovery. In order to prevent the need for retransmission when errors are repaired at the receiving end, FEC entails adding redundant information to the transmitted data. TCP can recover from packet loss without waiting for retransmissions by using FEC [254], which is advantageous on high-speed networks where delays are more apparent. On the other hand, the TCP protocol has an extension called SACK that enables the recipient to notify the sender of the packets that were successfully received. SACK reduces needless retransmissions and boosts overall efficiency by allowing the sender to retransmit only the specific lost packets instead of the full window of data, in contrast to standard acknowledgment systems [255]. According to [256], TCP Fast Retransmit is a mechanism that enables the sender to retransmit a lost packet based on the detection of duplicate acknowledgments. When the sender receives three duplicate acknowledgments for a particular packet (indicating that subsequent packets were successfully received), it assumes that the original packet was lost and initiates a fast retransmit, minimizing the time spent waiting for a timeout. Coupled with Fast Recovery, this mechanism allows TCP to quickly recover from packet loss and maintain a more consistent flow of data.

13.3. Window Size Limitations

TCP Window Scaling, TCP SACK and Use of High Performance TCP Variants (e.g., BBR) are the probable solutions. TCP Window Scaling makes it possible to raise the TCP window size above the customary 64 KB limit [257]. The bandwidth-delay product (BDP) in high-speed networks can lead to a lot of data being transferred. Window scaling makes it possible for the TCP window to represent a bigger BDP, improving bandwidth consumption and lessening the effect of throughput constraints. An addition to the TCP protocol called TCP SACK enables the recipient to acknowledge several non-contiguous data blocks. SACK increases TCP's efficiency [258] in high-speed networks by giving it more latitude when it comes to recognizing received data. In the event of a packet loss, it permits the sender to retransmit only the absent segments, negating the need to retransmit the complete data window. On the other hand, High-performance TCP variants, such as Bottleneck Bandwidth and Round-trip propagation time (BBR), are designed to optimize data transfer in high-speed networks [259]. BBR, for instance, employs a model that estimates the available bandwidth and adjusts the sending rate accordingly. It dynamically adapts to changing network conditions, providing better throughput and reducing the impact of limitations associated with traditional TCP congestion control algorithms.

13.4. Mitigating TCP Slow Start issues

TCP initial window size increase, congestion control algorithms modification, as well as Explicit Signaling or Explicit Congestion Notification (ECN) are some of the solutions here. Increasing the size of the initial congestion window is one simple strategy. The quantity of data that can be transferred before getting an acknowledgment is determined by the congestion window. By raising the initial window size, TCP can start with more data in high-speed networks, which shortens the time it takes to attain the ideal data rate [260]. Modifying the congestion control algorithms used by TCP is another solution. Traditional TCP slow start uses an additive increase, meaning it adds a fixed amount to the congestion window for each round-trip time [261]. In high-speed networks, a more aggressive increase may be beneficial. Some proposals suggest using a multiplicative increase or a hybrid approach to accelerate the growth of the congestion window. On the other hand, ECN is a mechanism that allows routers to signal congestion to the endpoints without dropping packets [262]. In high-speed networks, ECN can be used to provide early feedback to the sender about network congestion, allowing TCP to react more quickly and adjust its congestion window accordingly. This can help in avoiding unnecessary slow start phases.

13.5. Congestion Control Challenges

TCP BBR, TCP CUBIC and TCP Vegas are the feasible solutions. TCP BBR is a congestion control algorithm developed by Google. It focuses on estimating the available bandwidth and round-trip time to optimize data transmission [263]. BBR is designed to handle both low and high-speed networks effectively. It uses a model that estimates the bandwidth-delay product, helping it adapt to varying network conditions. The algorithm aims to maintain high throughput while being responsive to network congestion, making it well-suited for high-speed networks.TCP CUBIC is another congestion control algorithm designed to improve TCP performance [264] in high-speed and long-distance networks. Unlike traditional TCP variants, CUBIC uses a cubic growth function to increase the congestion window size during the slow start and congestion avoidance phases. This allows it to make more efficient use of available bandwidth in high-speed networks. CUBIC has been included in the Linux kernel and is widely used to address congestion control challenges in modern networks [266]. TCP Vegas is an alternative congestion control algorithm that focuses on reducing network latency by proactively detecting congestion before it causes packet loss. Instead of relying solely on packet loss as an indicator of congestion, Vegas monitors the round-trip time and estimates the ideal congestion window to achieve maximum throughput without causing congestion [267]. Vegas aims to provide better performance in high-speed networks by preventing unnecessary packet loss and optimizing the utilization of available bandwidth.

13.6. Techniques to mitigate Queueing Delays

Active Queue Management (AQM), TCP Variants and Enhancements, as well as Quality of Service (QoS) Policies are some of the solutions for this problem. AQM mechanisms, such as Random Early Detection (RED) or Explicit Congestion Notification (ECN), aim to proactively manage queue lengths and signal congestion before it becomes severe [268]. This helps prevent excessive queueing delays and bufferbloat. On the other hand TCP variants like TCP BBR are designed to handle queueing delays more efficiently in high-speed networks. These variants often employ advanced algorithms to optimize throughput and reduce latency. As explained in [269], QoS policies can be implemented to prioritize certain types of traffic and reduce the impact of queueing delays on critical applications or services.

13.7. Techniques to mitigate Bufferbloat

Some of the basic solutions include AQM, Buffer Size Optimization, as well as Traffic Shaping and Policing. AQM mechanisms, such as Random Early Detection (RED) or Explicit Congestion Notification (ECN), aim to actively manage queue lengths and prevent excessive buffering [270]. These mechanisms can help reduce bufferbloat and improve the responsiveness of TCP. On the other hand, proper sizing of buffers based on network characteristics [271] and traffic patterns is essential. Avoiding excessively large buffers helps prevent bufferbloat and ensures more responsive TCP performance [272]. Similarly, implementing traffic shaping and policing mechanisms can help control the rate of incoming traffic, preventing excessive queuing and bufferbloat [273].

13.8. Inefficient window scaling

TCP Window Scaling, SACK and use of high performance TCP variants are some of the probable solutions to this problem. TCP Window Scaling is a mechanism that allows the TCP window size to be increased beyond the traditional limit of 64 KB. In high-speed networks, the bandwidth-delay product (BDP) may result in a large amount of data in transit [274]. Window scaling enables the TCP window to represent a larger BDP, allowing for better utilization of available bandwidth and reducing the impact of limitations on throughput. On the other hand, TCP SACK is an extension to the TCP protocol that allows the receiver to acknowledge multiple non-contiguous blocks of data. SACK enhances the efficiency of TCP in high-speed networks by providing more flexibility in acknowledging received data [275]. It enables the sender to retransmit only the missing segments, reducing the need to retransmit the entire window of data in the case of packet loss. According to [276], high-performance TCP variants, such as BBR, are designed to optimize data transfer in high-speed networks. BBR, for instance, employs a model that estimates the available bandwidth and adjusts the sending rate accordingly. It dynamically adapts to changing network conditions [277], providing better throughput and reducing the impact of limitations associated with traditional TCP congestion control algorithms.

13.9. Head of Line Blocking

This problem can be addressed by techniques such as SACK, FEC, TCP Variants and Improvements, as well as QoS) Policies are some of the recommended solutions. SACK is an extension to TCP that allows the receiver to acknowledge non-contiguous blocks of data [278], enabling more efficient recovery from packet loss and reducing the likelihood of HoL blocking. On the other hand, FEC mechanisms can be employed to proactively correct errors without the need for retransmission, minimizing the impact of packet loss on subsequent data packets [279], [280]. As explained in [281] and [282], some TCP variants, such as TCP New Reno or TCP BBR, incorporate enhancements to address issues related to congestion control, retransmission, and flow control, improving performance in high-speed networks. Finally, implementing QoS policies to prioritize time-sensitive traffic, such as real-time applications, can help minimize the impact of HoL blocking on critical services [283], [284]. By assigning higher priority to packets associated with real-time applications, QoS mechanisms ensure that these packets are processed and delivered with minimal delay, even in the presence of congestion or buffering issues. This prioritization reduces the likelihood of HoL blocking, where delayed packets hinder the transmission of subsequent packets, thus ensuring timely delivery and minimizing latency for critical services. Additionally, QoS policies can allocate sufficient network resources and bandwidth to accommodate the requirements of real-time applications, further enhancing their performance and reliability in dynamic network environments.

14. Conclusion

This study has examined the complex interactions that occur between high-speed networks and the Transmission Control Protocol (TCP), providing insight into security and performance issues. After a thorough analysis, we have identified the difficulties that TCP deployment presents in settings with high data throughput. Our results highlight the fine tuning that must be done between the requirements of high-speed networks and TCP's intrinsic design. Even though TCP has shown to be robust in conventional environments, as network speeds increase, its performance quirks become more noticeable. In addition to identifying constraints and bottlenecks, this study investigated possible improvements to address these problems and maximize TCP in high-speed situations. In addition, our investigation into TCP security issues in high-speed networks has uncovered vulnerabilities that require close attention. Threats against networks change along with them. The significance of putting strong security measures in place to protect data integrity and confidentiality in contexts with rapid transmission has been highlighted by our investigation. To sum up, this study offers insightful information that advances the current discussion on networking protocol development. Through recognition and resolution of TCP's performance and security complexities in high-speed networks, we open the door to a more robust and effective communication infrastructure in the digital era.

References

[1] Tang J, Chen M, Chen H, Zhao S, Huang Y. A new dynamic security defense system based on TCP_REPAIR and deep learning. Journal of Cloud Computing. 2023 Feb 14;12(1):21.

[2] Li Y, Chen L, Su L, Zhao K, Wang J, Yang Y, Ge N. Pepesc: A TCP performance enhancing proxy for non-terrestrial networks. IEEE Transactions on Mobile Computing. 2023 Apr 24.

[3] Talau M, Herek TA, Fonseca M, Wille EC. Improving TCP performance over a common IoT scenario using the Early Window Tailoring method. Computer Networks. 2023 Oct 1;234:109875.

[4] Liu Z, Liang T, Wang W, Sun R, Li S. Design and Implementation of a Lightweight Security-Enhanced Scheme for Modbus TCP Protocol. Security and Communication Networks. 2023 Apr 13;2023.

[5] Hussain SZ, Parween S. Analysis of TCP issues and their possible solutions in the internet of things. Int. Arab J. Inf. Technol.. 2023 Mar 1;20(2):206-14.

[6] Hussien ZA, Abdulmalik HA, Hussain MA, Nyangaresi VO, Ma J, Abduljabbar ZA, Abduljaleel IQ. Lightweight Integrity Preserving Scheme for Secure Data Exchange in Cloud-Based IoT Systems. Applied Sciences. 2023 Jan;13(2):691.

[7] Kurose J, Ross K. Computer networks: A top down approach featuring the internet. Peorsoim Addison Wesley. 2010.

[8] Sangodoyin A, Modu B, Awan I, Disso JP. An approach to detecting distributed denial of service attacks in software defined networks. In2018 IEEE 6th International Conference on Future Internet of Things and Cloud (FiCloud) 2018 Aug 6 (pp. 436-443). IEEE.

[9] Lin J, Zhang X, Gao X, Kang P, Zhou Y, Ouyang Y, Feng T. Packet Reordering in the Era of 6G: Techniques, Challenges, and Applications. Electronics. 2023 Jul 10;12(14):3023.

[10] Kharat P, Kulkarni M. Congestion controlling schemes for high-speed data networks: A survey. Journal of High Speed Networks. 2019 Jan 1;25(1):41-60.

[11] Çelebi M, Özbilen A, Yavanoğlu U. A comprehensive survey on deep packet inspection for advanced network traffic analysis: issues and challenges. Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi. 2023 Jan 1;12(1):1-29.

[12] Rashed AN, Ahammad SH, Daher MG, Sorathiya V, Siddique A, Asaduzzaman S, Rehana H, Dutta N, Patel SK, Nyangaresi VO, Jibon RH. Spatial single mode laser source interaction with measured pulse based parabolic index multimode fiber. Journal of Optical Communications. 2022 Jun 21.

[13] Muzammil MB, Bilal M, Ajmal S, Shongwe SC, Ghadi YY. Unveiling Vulnerabilities of Web Attacks Considering Man in the Middle Attack and Session Hijacking. IEEE Access. 2024 Jan 5.

[14] Aslan Ö, Aktuğ SS, Ozkan-Okay M, Yilmaz AA, Akin E. A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions. Electronics. 2023 Mar 11;12(6):1333.

[15] Feng X, Li Q, Sun K, Yang Y, Xu K. Man-in-the-middle attacks without rogue ap: When wpas meet icmp redirects. In2023 IEEE Symposium on Security and Privacy (SP) 2023 May 21 (pp. 3162-3177). IEEE.

[16] Chavoshi H, Salasi A, Payam O, Khaloozadeh H. Man-in-the-Middle Attack Against a Network Control System: Practical Implementation and Detection. In2023 IEEE 64th International Scientific Conference on Information Technology and Management Science of Riga Technical University (ITMS) 2023 Oct 5 (pp. 1-6). IEEE.

[17] Nyangaresi VO, Al-Joboury IM, Al-sharhanee KA, Najim AH, Abbas AH, Hariz HM. A Biometric and Physically Unclonable Function-Based Authentication Protocol for Payload Exchanges in Internet of Drones. e-Prime-Advances in Electrical Engineering, Electronics and Energy. 2024 Feb 23:100471.

[18] Iftikhar A, Qureshi KN, Shiraz M, Albahli S. Security, trust and privacy risks, responses, and solutions for high-speed smart cities networks: A systematic literature review. Journal of King Saud University-Computer and Information Sciences. 2023 Oct 13:101788.

[19] Kommineni KK, Prasad A. A Review on Privacy and Security Improvement Mechanisms in MANETs. International Journal of Intelligent Systems and Applications in Engineering. 2024;12(2):90-9.

[20] Kaluvakuri S, Lal K. Networking Alchemy: Demystifying the Magic behind Seamless Digital Connectivity. International Journal of Reciprocal Symmetry and Theoretical Physics. 2017;4:20-8.

[21] Jahanbakht M, Xiang W, Hanzo L, Azghadi MR. Internet of underwater things and big marine data analytics—a comprehensive survey. IEEE Communications Surveys & Tutorials. 2021 Jan 20;23(2):904-56.

[22] Bhattacharjee S. Practical Industrial Internet of Things security: A practitioner's guide to securing connected industries. Packt Publishing Ltd; 2018 Jul 30.

[23] Mohammed MA, Hussain MA, Oraibi ZA, Abduljabbar ZA, Nyangaresi VO. Secure Content Based Image Retrieval System Using Deep Learning. J. Basrah Res.(Sci.). 2023 Dec 30;49(2):94-111.

[24] Safari F, Kunze H, Ernst J, Gillis D. A novel cross-layer adaptive fuzzy-based ad hoc on-demand distance vector routing protocol for MANETs. IEEE Access. 2023 May 18.

[25] Radwan A, Chi HR. Towards Cell-Free Networking: Analytical Study of Ultra-Dense On-Demand Small Cell Deployment for Internet of Things. In2023 International Wireless Communications and Mobile Computing (IWCMC) 2023 Jun 19 (pp. 1202-1207). IEEE.

[26] Wang G, Wu J, Trik M. A novel approach to reduce video traffic based on understanding user demand and D2D communication in 5G networks. IETE Journal of Research. 2023 Nov 22:1-7.

[27] Kaddoura S, Haraty RA, Al Jahdali S, Assi M. SDODV: A smart and adaptive on-demand distance vector routing protocol for MANETs. Peer-to-Peer Networking and Applications. 2023 Sep;16(5):2325-48.

[28] Zaki Rashed AN, Ahammad SH, Daher MG, Sorathiya V, Siddique A, Asaduzzaman S, Rehana H, Dutta N, Patel SK, Nyangaresi VO, Jibon RH. Signal propagation parameters estimation through designed multi layer fibre with higher dominant modes using OptiFibre simulation. Journal of Optical Communications. 2022 Jun 23(0).

[29] Dwivedi A, Tiwari P, Pandey S. Transmission Control Protocol/Internet Protocol: Security Issues & Solution. International Journal of Scientific Research in Modern Science and Technology. 2023 Feb 28;2(2):01-8.

[30] Li J, Wu Y, Li Y, Zhang Z, Fouad H, Altameem T. A Network Security Prediction Method Based on Attack Defense Tree. Journal of Nanoelectronics and Optoelectronics. 2023 Mar 1;18(3):357-66.

[31] Sudar KM, Deepalakshmi P, Singh A, Srinivasu PN. TFAD: TCP flooding attack detection in software-defined networking using proxy-based and machine learning-based mechanisms. Cluster Computing. 2023 Apr;26(2):1461-77.

[32] Zhang H, Min Y, Liu S, Tong H, Li Y, Lv Z. Improve the Security of Industrial Control System: A Fine-Grained Classification Method for DoS Attacks on Modbus/TCP. Mobile Networks and Applications. 2023 Apr;28(2):839-52.

[33] Abou El Houda Z, Brik B, Senouci SM. A novel iot-based explainable deep learning framework for intrusion detection systems. IEEE Internet of Things Magazine. 2022 Jun;5(2):20-3.

[34] Nyangaresi VO. Extended Chebyshev Chaotic Map Based Message Verification Protocol for Wireless Surveillance Systems. InComputer Vision and Robotics: Proceedings of CVR 2022 2023 Apr 28 (pp. 503-516). Singapore: Springer Nature Singapore.

[35] Pan Y, Rossow C. TCP Spoofing: Reliable Payload Transmission Past the Spoofed TCP Handshake. In2024 IEEE Symposium on Security and Privacy (SP) 2024 Feb 1 (pp. 179-179). IEEE Computer Society.

[36] Abdulkarim MK, Adebayo OS, Abdulhamid SM, Bashir SA. Systematic Review of Session Hijacking Attacks on 5G Network. 2nd IOU Conference on Research and Integrated Sciences (IOUCRIS) 2022. International Open University,.

[37] Wang Z, Feng X, Li Q, Sun K, Yang Y, Li M, Xu K. Off-Path TCP Hijacking in Wi-Fi Networks: A Packet-Size Side Channel Attack. arXiv preprint arXiv:2402.12716. 2024 Feb 20.

[38] Li S, Shi S, Xiao Y, Zhang C, Hou YT, Lou W. Bijack: Breaking Bitcoin Network with TCP Vulnerabilities. InEuropean Symposium on Research in Computer Security 2023 Sep 25 (pp. 306-326). Cham: Springer Nature Switzerland.

[39] Abduljabbar ZA, Nyangaresi VO, Jasim HM, Ma J, Hussain MA, Hussien ZA, Aldarwish AJ. Elliptic Curve Cryptography-Based Scheme for Secure Signaling and Data Exchanges in Precision Agriculture. Sustainability. 2023 Jun 28;15(13):10264.

[40] Mittal M, Kumar K, Behal S. DDoS-AT-2022: a distributed denial of service attack dataset for evaluating DDoS defense system. Proceedings of the Indian National Science Academy. 2023 Jun;89(2):306-24.

[41] Vedula V, Lama P, Boppana RV, Trejo LA. On the detection of low-rate denial of service attacks at transport and application layers. Electronics. 2021 Aug 30;10(17):2105.

[42] Batchu RK, Seetha H. An integrated approach explaining the detection of distributed denial of service attacks. Computer Networks. 2022 Oct 24;216:109269.

[43] Vijayalakshmi S, Bose S, Logeswari G, Anitha TJ. Hybrid defense mechanism against malicious packet dropping attack for MANET using game theory. Cyber security and applications. 2023 Dec 1;1:100011.

[44] Kumari P, Jain AK. A comprehensive study of DDoS attacks over IoT network and their countermeasures. Computers & Security. 2023 Apr 1;127:103096.

[45] Sánchez-Patiño N, Gallegos-Garcia G, Rivero-Angeles ME. Teletraffic Analysis of DoS and Malware Cyber Attacks on P2P Networks under Exponential Assumptions. Applied Sciences. 2023 Apr 6;13(7):4625.

[46] Nyangaresi VO. Provably secure authentication protocol for traffic exchanges in unmanned aerial vehicles. High-Confidence Computing. 2023 Sep 15:100154.

[47] Haseeb-Ur-Rehman RM, Aman AH, Hasan MK, Ariffin KA, Namoun A, Tufail A, Kim KH. High-Speed Network DDoS Attack Detection: A Survey. Sensors. 2023 Aug 1;23(15):6850.

[48] Mahajan N, Chauhan A, Kumar H, Kaushal S, Sangaiah AK. A deep learning approach to detection and mitigation of distributed denial of service attacks in high availability intelligent transport systems. Mobile Networks and Applications. 2022 Aug;27(4):1423-43.

[49] Kaur Chahal J, Bhandari A, Behal S. Distributed denial of service attacks: a threat or challenge. New Review of Information Networking. 2019 Jan 2;24(1):31-103.

[50] Chen M, Chen J, Wei X, Chen B. Is low-rate distributed denial of service a great threat to the Internet?. IET Information Security. 2021 Sep;15(5):351-63.

[51] Suhag A, Daniel A. Study of statistical techniques and artificial intelligence methods in distributed denial of service (DDOS) assault and defense. Journal of Cyber Security Technology. 2023 Jan 2;7(1):21-51.

[52] Omollo VN, Musyoki S. Global Positioning System Based Routing Algorithm for Adaptive Delay Tolerant Mobile Adhoc Networks. International Journal of Computer and Communication System Engineering. 2015 May 11; 2(3): 399-406.

[53] Vishwakarma R, Jain AK. A survey of DDoS attacking techniques and defence mechanisms in the IoT network. Telecommunication systems. 2020 Jan;73(1):3-25.

[54] Alzahrani RJ, Alzahrani A. Security analysis of DDoS attacks using machine learning algorithms in networks traffic. Electronics. 2021 Nov 25;10(23):2919.

[55] Ahuja N, Singal G, Mukhopadhyay D, Kumar N. Automated DDOS attack detection in software defined networking. Journal of Network and Computer Applications. 2021 Aug 1;187:103108.

[56] Novaes MP, Carvalho LF, Lloret J, Proença Jr ML. Adversarial Deep Learning approach detection and defense against DDoS attacks in SDN environments. Future Generation Computer Systems. 2021 Dec 1;125:156-67.

[57] Nyangaresi VO. Target Tracking Area Selection and Handover Security in Cellular Networks: A Machine Learning Approach. InProceedings of Third International Conference on Sustainable Expert Systems: ICSES 2022 2023 Feb 23 (pp. 797-816). Singapore: Springer Nature Singapore.

[58] Cheema A, Tariq M, Hafiz A, Khan MM, Ahmad F, Anwar M. Prevention techniques against distributed denial of service attacks in heterogeneous networks: A systematic review. Security and Communication Networks. 2022 May 20;2022:1-5.

[59] Rios VD, Inacio PR, Magoni D, Freire MM. Detection and mitigation of low-rate denial-of-service attacks: A survey. IEEE Access. 2022 Jul 15;10:76648-68.

[60] Nurwarsito H, Nadhif MF. DDoS attack early detection and mitigation system on SDN using random forest algorithm and Ryu framework. In2021 8th International Conference on Computer and Communication Engineering (ICCCE) 2021 Jun 22 (pp. 178-183). IEEE.

[61] Allakany A. Cost-Efficient Method for Detecting and Mitigating DDOS Attacks in SDN Based Networks. Kafrelsheikh Journal of Information Sciences. 2023 Nov 1;4(2):1-0.

[62] Shi K, Cai X, She K, Wen S, Zhong S, Park P, Kwon OM. Stability analysis and security-based event-triggered mechanism design for TS fuzzy NCS with traffic congestion via DoS attack and its application. IEEE Transactions on Fuzzy Systems. 2023 Mar 28.

[63] Omollo VN, Musyoki S. Blue bugging Java Enabled Phones via Bluetooth Protocol Stack Flaws. International Journal of Computer and Communication System Engineering. 2015 Jun 9, 2 (4):608-613.

[64] Chaudhary D, Bhushan K, Gupta BB. Survey on DDoS attacks and defense mechanisms in cloud and fog computing. International Journal of E-Services and Mobile Applications (IJESMA). 2018 Jul 1;10(3):61-83.

[65] Tan L, Huang K, Peng G, Chen G. Stability of TCP/AQM networks under DDoS attacks with design. IEEE Transactions on Network Science and Engineering. 2020 Jul 27;7(4):3042-56.

[66] Zhijun W, Wenjing L, Liang L, Meng Y. Low-rate DoS attacks, detection, defense, and challenges: a survey. IEEE access. 2020 Feb 27;8:43920-43.

[67] Adedeji KB, Abu-Mahfouz AM, Kurien AM. DDoS attack and detection methods in internet-enabled networks: Concept, research perspectives, and challenges. Journal of Sensor and Actuator Networks. 2023 Jul 6;12(4):51.

[68] Nyangaresi VO, Moundounga AR. Secure data exchange scheme for smart grids. In2021 IEEE 6th International Forum on Research and Technology for Society and Industry (RTSI) 2021 Sep 6 (pp. 312-316). IEEE.

[69] Chitre P, Sriramulu S. DDoS Attack, a Threat to IoT Devices in the High-Speed Networks—An Overview. InInternational Conference on Network Security and Blockchain Technology 2023 Mar 24 (pp. 205-215). Singapore: Springer Nature Singapore.

[70] Nayak G, Mishra A, Samal U, Mishra BK. Depth analysis on DoS & DDoS attacks. Wireless Communication Security. 2022 Dec 7:159-82.

[71] Shukla P, Krishna CR, Patil NV. Iot traffic-based DDoS attacks detection mechanisms: A comprehensive review. The Journal of Supercomputing. 2023 Dec 19:1-58.

[72] Lorincz J, Klarin Z, Ožegović J. A comprehensive overview of TCP congestion control in 5G networks: Research challenges and future perspectives. Sensors. 2021 Jun 30;21(13):4510.

[73] Kushwaha V, Gupta R. Congestion control for high-speed wired network: A systematic literature review. Journal of Network and Computer Applications. 2014 Oct 1;45:62-78.

[74] Mohammad Z, Nyangaresi V, Abusukhon A. On the Security of the Standardized MQV Protocol and Its Based Evolution Protocols. In2021 International Conference on Information Technology (ICIT) 2021 Jul 14 (pp. 320-325). IEEE.

[75] Kanagarathinam MR, Singh S, Sandeep I, Kim H, Maheshwari MK, Hwang J, Roy A, Saxena N. NexGen D-TCP: Next generation dynamic TCP congestion control algorithm. IEEE Access. 2020 Sep 7;8:164482-96.

[76] Saedi T, El-Ocla H. TCP CERL+: Revisiting TCP congestion control in wireless networks with random loss. Wireless Networks. 2021 Jan;27:423-40.

[77] Haile H, Grinnemo KJ, Ferlin S, Hurtig P, Brunstrom A. End-to-end congestion control approaches for high throughput and low delay in 4G/5G cellular networks. Computer Networks. 2021 Feb 26;186:107692.

[78] Verma LP, Sharma VK, Kumar M, Kanellopoulos D. A novel delay-based adaptive congestion control TCP variant. Computers and Electrical Engineering. 2022 Jul 1;101:108076.

[79] Ali MH, Öztürk S. Efficient congestion control in communications using novel weighted ensemble deep reinforcement learning. Computers and Electrical Engineering. 2023 Sep 1;110:108811.

[80] Eid MM, Arunachalam R, Sorathiya V, Lavadiya S, Patel SK, Parmar J, Delwar TS, Ryu JY, Nyangaresi VO, Zaki Rashed AN. QAM receiver based on light amplifiers measured with effective role of optical coherent duobinary transmitter. Journal of Optical Communications. 2022 Jan 17(0).

[81] Mishra TK, Sahoo KS, Bilal M, Shah SC, Mishra MK. Adaptive congestion control mechanism to enhance TCP performance in cooperative IoV. IEEE Access. 2023 Jan 23;11:9000-13.

[82] Na W, Lakew DS, Lee J, Cho S. Congestion control vs. link failure: TCP behavior in mmWave connected vehicular networks. Future Generation Computer Systems. 2019 Dec 1;101:1213-22.

[83] Zeng G, Bai W, Chen G, Chen K, Han D, Zhu Y, Cui L. Congestion control for cross-datacenter networks. IEEE/ACM Transactions on Networking. 2022 Apr 5;30(5):2074-89.

[84] Pan W, Tan H, Li X, Li X. Improved RTT fairness of BBR congestion control algorithm based on adaptive congestion window. Electronics. 2021 Mar 6;10(5):615.

[85] Nyangaresi VO. A Formally Verified Authentication Scheme for mmWave Heterogeneous Networks. Inthe 6th International Conference on Combinatorics, Cryptography, Computer Science and Computation (605-612) 2021.

[86] Wei W, Xue K, Han J, Wei DS, Hong P. Shared bottleneck-based congestion control and packet scheduling for multipath TCP. IEEE/ACM Transactions on Networking. 2020 Feb 13;28(2):653-66.

[87] Lim C. Improving congestion control of TCP for constrained IoT networks. Sensors. 2020 Aug 24;20(17):4774.

[88] Shao J, Li M, Li X, Liu G, Liu S, Liu B, Xu Y. RaceCC: A rapidly converging explicit congestion control for datacenter networks. Journal of Network and Computer Applications. 2023 Aug 1;217:103673.

[89] Zhong X, Zhang J, Zhang Y, Guan Z, Wan Z. PACC: Proactive and accurate congestion feedback for RDMA congestion control. InIEEE INFOCOM 2022-IEEE Conference on Computer Communications 2022 May 2 (pp. 2228-2237). IEEE.

[90] Freitas E, de Oliveira Filho AT, do Carmo PR, Sadok D, Kelner J. A survey on accelerating technologies for fast network packet processing in Linux environments. Computer Communications. 2022 Dec 1;196:148-66.

[91] Poorzare R, Augé AC. Challenges on the way of implementing TCP over 5G networks. IEEE access. 2020 Sep 24;8:176393-415.

[92] Feng J, Ouyang Z, Xu L, Ramamurthy B. Packet reordering in high-speed networks and its impact on high-speed TCP variants. Computer Communications. 2009 Jan 23;32(1):62-8.

[93] Mutlaq KA, Nyangaresi VO, Omar MA, Abduljabbar ZA, Abduljaleel IQ, Ma J, Al Sibahee MA. Low complexity smart grid security protocol based on elliptic curve cryptography, biometrics and hamming distance. Plos one. 2024 Jan 23;19(1):e0296781.

[94] Huang J, Lyu W, Li W, Wang J, He T. Mitigating packet reordering for random packet spraying in data center networks. IEEE/ACM Transactions on Networking. 2021 Feb 10;29(3):1183-96.

[95] Handley M, Raiciu C, Agache A, Voinescu A, Moore AW, Antichi G, Wójcik M. Re-architecting datacenter networks and stacks for low latency and high performance. InProceedings of the Conference of the ACM Special Interest Group on Data Communication 2017 Aug 7 (pp. 29-42).

[96] Briscoe B, Brunstrom A, Petlund A, Hayes D, Ros D, Tsang J, Gjessing S, Fairhurst G, Griwodz C, Welzl M. Reducing internet latency: A survey of techniques and their merits. IEEE Communications Surveys & Tutorials. 2014 Nov 26;18(3):2149-96.

[97] Yan B, Liu Q, Shen J, Liang D, Zhao B, Ouyang L. A survey of low-latency transmission strategies in software defined networking. Computer Science Review. 2021 May 1;40:100386.

[98] Srivastava A, Fund F, Panwar SS. Some of the Internet may be heading towards BBR dominance: an experimental study. InIEEE INFOCOM 2023-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS) 2023 May 20 (pp. 1-7). IEEE.

[99] Al Sibahee MA, Ma J, Nyangaresi VO, Abduljabbar ZA. Efficient Extreme Gradient Boosting Based Algorithm for QoS Optimization in Inter-Radio Access Technology Handoffs. In2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA) 2022 Jun 9 (pp. 1-6). IEEE.

[100] Vladimirov S, Vybornova A, Muthanna A, Koucheryavy A, Abd El-Latif AA. Network Coding Datagram Protocol for TCP/IP Networks. IEEE Access. 2023 Apr 11.

[101] Kim D, Moon Y, Hwang J, Park K. FlexCP: A Scalable Multipath TCP Proxy for Cellular Networks. Proceedings of the ACM on Networking. 2023 Nov 28;1(CoNEXT3):1-21.

[102] Mahmoodi Khaniabadi S, Javadpour A, Gheisari M, Zhang W, Liu Y, Sangaiah AK. An intelligent sustainable efficient transmission internet protocol to switch between User Datagram Protocol and Transmission Control Protocol in IoT computing. Expert Systems. 2023 Jun;40(5):e13129.

[103] Abdullah SM, Farag MS, Abdul-Kader H, Youssef SE. Improving the TCP Newreno Congestion Avoidance Algorithm on 5G Networks. J. Commun.. 2023 Apr;18(4):228-35.

[104] Sundararajan JK, Shah D, Médard M, Jakubczak S, Mitzenmacher M, Barros J. Network coding meets TCP: Theory and implementation. Proceedings of the IEEE. 2011 Jan 13;99(3):490-512.

[105] Al Sibahee MA, Nyangaresi VO, Abduljabbar ZA, Luo C, Zhang J, Ma J. Two-Factor Privacy Preserving Protocol for Efficient Authentication in Internet of Vehicles Networks. IEEE Internet of Things Journal. 2023 Dec 7.

[106] Kaur S, Kaur G, Shabaz M. A secure two-factor authentication framework in cloud computing. Security and Communication Networks. 2022 Mar 12;2022:1-9.

[107] AlQahtani AA, Alshayeb T, Nabil M, Patooghy A. Leveraging Machine Learning for Wi-Fi-based Environmental Continuous Two-Factor Authentication. IEEE Access. 2024 Jan 19.

[108] Liu K, Zhou Z, Cao Q, Xu G, Wang C, Gao Y, Zeng W, Xu G. A Robust and Effective Two-Factor Authentication (2FA) Protocol Based on ECC for Mobile Computing. Applied Sciences. 2023 Mar 30;13(7):4425.

[109] Wang Q, Wang D, Cheng C, He D. Quantum2fa: efficient quantum-resistant two-factor authentication scheme for mobile devices. IEEE Transactions on Dependable and Secure Computing. 2021 Nov 22;20(1):193-208.

[110] Qian Z, Mao ZM. Off-path TCP sequence number inference attack-how firewall middleboxes reduce security. In2012 IEEE Symposium on Security and Privacy 2012 May 20 (pp. 347-361). IEEE.

[111] Satish Kumar V, Dutta T, Sur A, Nandi S. Secure network steganographic scheme exploiting TCP sequence numbers. InAdvances in Network Security and Applications: 4th International Conference, CNSA 2011, Chennai, India, July 15-17, 2011 4 2011 (pp. 281-291). Springer Berlin Heidelberg.

[112] Nyangaresi VO. Masked Symmetric Key Encrypted Verification Codes for Secure Authentication in Smart Grid Networks. In2022 4th Global Power, Energy and Communication Conference (GPECOM) 2022 Jun 14 (pp. 427-432). IEEE.

[113] Sundberg S, Brunstrom A, Ferlin-Reiter S, Høiland-Jørgensen T, Brouer JD. Efficient continuous latency monitoring with eBPF. InInternational Conference on Passive and Active Network Measurement 2023 Mar 10 (pp. 191-208). Cham: Springer Nature Switzerland.

[114] Hughes LE. SSL and TLS. InPro Active Directory Certificate Services: Creating and Managing Digital Certificates for Use in Microsoft Networks 2022 Mar 19 (pp. 155-175). Berkeley, CA: Apress.

[115] Yaseen M, Kamel MB, Ligeti P. Security Analysis and Deployment Measurement of Transport Layer Security Protocol. InRecent Innovations in Computing: Proceedings of ICRIC 2021, Volume 2 2022 Apr 16 (pp. 725-739). Singapore: Springer Singapore.

[116] Sharma M, Sharma M, Sharma N. A Cutting-Edge AI-and-IoT-Powered Cyber Secured Intrusion Detection System. In2023 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES) 2023 Dec 14 (pp. 1-5). IEEE.

[117] ElKashlan M, Aslan H, Said Elsayed M, Jurcut AD, Azer MA. Intrusion detection for electric vehicle charging systems (evcs). Algorithms. 2023 Jan 31;16(2):75.

[118] Al-Haddad R, Velazquez ES, Fatima A, Winckles A. A novel traffic shaping algorithm for SDN-sliced networks using a new WFQ technique. International Journal of Advanced Computer Science and Applications. 2021;12(1).

[119] Ahmad AY, Verma N, Sarhan N, Awwad EM, Arora A, Nyangaresi VO. An IoT and Blockchain-Based Secure and Transparent Supply Chain Management Framework in Smart Cities Using Optimal Queue Model. IEEE Access. 2024 Mar 18.

[120] Tran TV, Ahn H. Challenges of and solution to the control load of stateful firewall in software defined networks. Computer Standards & Interfaces. 2017 Nov 1;54:293-304.

[121] Anand A, Das S, Agarwal M, Inoue S. An optimal scheduling policy for upgraded software with updates. International Journal of Quality & Reliability Management. 2022 Feb 22;39(3):704-15.

[122] Camacho J, Pérez-Villegas A, García-Teodoro P, Maciá-Fernández G. PCA-based multivariate statistical network monitoring for anomaly detection. Computers & Security. 2016 Jun 1;59:118-37.

[123] Mdini M, Blanc A, Simon G, Barotin J, Lecoeuvre J. Monitoring the network monitoring system: Anomaly Detection using pattern recognition. In2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM) 2017 May 8 (pp. 983-986). IEEE.

[124] Reddy AV, Kumar AA, Venu N, Reddy RV. On optimization efficiency of scalability and availability of cloud-based software services using scale rate limiting algorithm. Measurement: Sensors. 2022 Dec 1;24:100468.

[125] Nyangaresi VO, Morsy MA. Towards privacy preservation in internet of drones. In2021 IEEE 6th International Forum on Research and Technology for Society and Industry (RTSI) 2021 Sep 6 (pp. 306-311). IEEE.

[126] Kumar Y, Kumar V. A Systematic Review on Intrusion Detection System in Wireless Networks: Variants, Attacks, and Applications. Wireless Personal Communications. 2023 Dec 23:1-58.

[127] Qaddoori SL, Ali QI. An embedded intrusion detection and prevention system for home area networks in advanced metering infrastructure. IET Information Security. 2023 May;17(3):315-34.

[128] Molnár M, Le DD, Perelló J, Solé-Pareta J, McArdle C. Multicast routing from a set of data centers in elastic optical networks. Optical Switching and Networking. 2019 Nov 1;34:35-46.

[129] Wong KS, Wan TC. Current state of multicast routing protocols for disruption tolerant networks: Survey and open issues. Electronics. 2019 Feb 1;8(2):162.

[130] Makhkamov B, Khasanov N. A survey of cache placement algorithms in content delivery networks. InE3S Web of Conferences 2023 (Vol. 458, p. 09004). EDP Sciences.

[131] Abduljabbar ZA, Nyangaresi VO, Jasim HM, Ma J, Hussain MA, Hussien ZA, Aldarwish AJ. Elliptic Curve Cryptography-Based Scheme for Secure Signaling and Data Exchanges in Precision Agriculture. Sustainability. 2023 Jun 28;15(13):10264.

[132] Ahmed MM, El-Hajjar A. A proactive approach to protect cloud computing environment against a distributed denial of service (DDoS) attack. InAI, Blockchain and Self-Sovereign Identity in Higher Education 2023 Jun 23 (pp. 243-278). Cham: Springer Nature Switzerland.

[133] Devi BK, Subbulakshmi T. Intrusion detection and prevention of DDoS attacks in cloud computing environment: a review on issues and current methods. International Journal of Cloud Computing. 2023;12(5):450-81.

[134] Paolucci F. Network service chaining using segment routing in multi-layer networks. Journal of Optical Communications and Networking. 2018 Jun 1;10(6):582-92.

[135] Rey WP, Rey KW. Towards a Redundant Internetwork Structure of Exterior and Interior Border Gateway Protocol (BGP) Sessions in an Enterprise Network. In2023 International Conference on Information Network and Computer Communications (INCC) 2023 Oct 27 (pp. 53-59). IEEE.

[136] Seliem M, Zahran A, Pesch D. Delay analysis of TSN based industrial networks with preemptive traffic using network calculus. In2023 IFIP Networking Conference (IFIP Networking) 2023 Jun 12 (pp. 1-9). IEEE.

[137] Nyangaresi VO, Ahmad M, Alkhayyat A, Feng W. Artificial neural network and symmetric key cryptography based verification protocol for 5G enabled Internet of Things. Expert Systems. 2022 Dec;39(10):e13126.

[138] Alshehri A, Khan N, Alowayr A, Alghamdi MY. Cyberattack Detection Framework Using Machine Learning and User Behavior Analytics. Computer Systems Science & Engineering. 2023 Feb 1;44(2).

[139] Irfan M, Gohar F, Sohail U, Jing Y. Information Security Framework Targeting DDOS attacks in Financial Institutes. IJLAI Transactions on Science and Engineering. 2023 Aug 21;1(01):1-52.

[140] Agrawal V, Wasnik P, Snekkenes EA. Factors Influencing the Participation of Information Security Professionals in Electronic Communities of Practice. InKMIS 2017 Nov (pp. 50-60).

[141] Praseed A, Thilagam PS. DDoS attacks at the application layer: Challenges and research perspectives for safeguarding web applications. IEEE Communications Surveys & Tutorials. 2018 Sep 16;21(1):661-85.

[142] Lakshminarayana S, Praseed A, Thilagam PS. Securing the IoT Application Layer from an MQTT Protocol Perspective: Challenges and Research Prospects. IEEE Communications Surveys & Tutorials. 2024 Mar 4.

[143] Al-Chaab W, Abduljabbar ZA, Abood EW, Nyangaresi VO, Mohammed HM, Ma J. Secure and Low-Complexity Medical Image Exchange Based on Compressive Sensing and LSB Audio Steganography. Informatica. 2023 May 31;47(6).

[144] Moura GC, Hesselman C, Schaapman G, Boerman N, De Weerdt O. Into the DDoS maelstrom: a longitudinal study of a scrubbing service. In2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW) 2020 Sep 7 (pp. 550-558). IEEE.

[145] Arsan T, Özbilen R. Bandwidth Allocation and Traffic Shaping in Mobile Broadband Networks using Deep Packet Inspection. International Journal of Computer Science and Information Security. 2016 Aug 1;14(8):1066.

[146] Ucar S, Hoh B, Oguchi K. Differential deviation based abnormal driving behavior detection. In2021 IEEE International Intelligent Transportation Systems Conference (ITSC) 2021 Sep 19 (pp. 1553-1558). IEEE.

[147] Muhammad A, Skorin-Kapov N, Furdek M. Manycast, anycast, and replica placement in optical inter-datacenter networks. Journal of Optical Communications and Networking. 2017 Dec 1;9(12):1161-71.

[148] Bhosale KS, Nenova M, Iliev G. The distributed denial of service attacks (DDoS) prevention mechanisms on application layer. In2017 13th International Conference on Advanced Technologies, Systems and Services in Telecommunications (TELSIKS) 2017 Oct 18 (pp. 136-139). IEEE.

[149] Birkinshaw C, Rouka E, Vassilakis VG. Implementing an intrusion detection and prevention system using software-defined networking: Defending against port-scanning and denial-of-service attacks. Journal of Network and Computer Applications. 2019 Jun 15;136:71-85.

[150] Nyangaresi VO. Privacy Preserving Three-factor Authentication Protocol for Secure Message Forwarding in Wireless Body Area Networks. Ad Hoc Networks. 2023 Apr 1;142:103117.

[151] Zhang T, Wang J, Huang J, Chen J, Pan Y, Min G. Tuning the aggressive TCP behavior for highly concurrent HTTP connections in intra-datacenter. IEEE/ACM Transactions on Networking. 2017 Oct 30;25(6):3808-22.

[152] Yu Y, Guo L, Liu Y, Zheng J, Zong YU. An efficient SDN-based DDoS attack detection and rapid response platform in vehicular networks. IEEE access. 2018 Jul 9;6:44570-9.

[153] Wagner D, Kopp D, Wichtlhuber M, Dietzel C, Hohlfeld O, Smaragdakis G, Feldmann A. United we stand: Collaborative detection and mitigation of amplification ddos attacks at scale. InProceedings of the 2021 ACM SIGSAC conference on computer and communications security 2021 Nov 12 (pp. 970-987).

[154] Agrawal N, Tapaswi S. Defense mechanisms against DDoS attacks in a cloud computing environment: State-of-the-art and research challenges. IEEE Communications Surveys & Tutorials. 2019 Aug 12;21(4):3769-95.

[155] Chiesa M, Kamisiński A, Rak J, Rétvári G, Schmid S. A survey of fast-recovery mechanisms in packet-switched networks. IEEE Communications Surveys & Tutorials. 2021 Mar 11;23(2):1253-301.

[156] Al Sibahee MA, Nyangaresi VO, Ma J, Abduljabbar ZA. Stochastic Security Ephemeral Generation Protocol for 5G Enabled Internet of Things. InIoT as a Service: 7th EAI International Conference, IoTaaS 2021, Sydney, Australia, December 13–14, 2021, Proceedings 2022 Jul 8 (pp. 3-18). Cham: Springer International Publishing.

[157] Usubütün U, Fund F, Panwar S. Do Switches Still Need to Deliver Packets in Sequence?. In2023 IEEE 24th International Conference on High Performance Switching and Routing (HPSR) 2023 Jun 5 (pp. 89-95). IEEE.

[158] Süzer AE, Oktal H. A comparison analysis on forward error correction technology: a future perspective for GNSS. Aircraft Engineering and Aerospace Technology. 2023 Jul 21;95(8):1311-20.

[159] Kumar V, Tyagi N. Device-centric data reordering and buffer management for mobile Internet using Multipath Transmission Control Protocol. International Journal of Communication Systems. 2021 Nov 25;34(17):e4973.

[160] Bruhn P, Kuehlewind M, Muehleisen M. Performance and improvements of TCP CUBIC in low-delay cellular networks. Computer Networks. 2023 Apr 1;224:109609.

[161] Jung J, Lee C, Baik J, Chung JM. Reveno: Rtt estimation based multipath tcp in 5g multi-rat networks. IEEE Transactions on Mobile Computing. 2022 May 26.

[162] Feng X, Li Q, Sun K, Xu K, Liu B, Zheng X, Yang Q, Duan H, Qian Z. PMTUD is not Panacea: Revisiting IP Fragmentation Attacks against TCP. InNDSS 2022 Apr.

[163] Nyakomitta PS, Nyangaresi VO, Ogara SO. Efficient authentication algorithm for secure remote access in wireless sensor networks. Journal of Computer Science Research. 2021 Aug;3(4):43-50.

[164] Ali I, Hong S, Park PK, Kim TY. Rethinking Explicit Congestion Notification: A Multilevel Congestion Feedback Perspective. InProceedings of the 34th edition of the Workshop on Network and Operating System Support for Digital Audio and Video 2024 Apr 15 (pp. 64-70).

[165] Fu L, Zhang H, Zhang Z, Li J, Guan H. FlowLever: Leverage Flow Director for Packet Dispatch Acceleration in NFV. IEEE Access. 2024 Mar 7.

[166] Kabirkhoo Z, Radpour M, Belostotski L. Tunable wideband high-order active analog delays with high delay-bandwidth product. IEEE Microwave and Wireless Technology Letters. 2023 Jun 13.

[167] Liu Y, Yang Z, Peng Y, Bi T, Jiang T. Bandwidth-Delay Product Based ACK Optimization Strategy for QUIC in Wi-Fi Networks. IEEE Internet of Things Journal. 2023 May 18.

[168] Yang J, Han J, Xue K, Wang Y, Li J, Xing Y, Yue H, Wei DS. TCCC: a throughput consistency congestion control algorithm for MPTCP in mixed transmission of long and short flows. IEEE Transactions on Network and Service Management. 2023 Feb 6.

[169] Nyangaresi VO, Petrovic N. Efficient PUF based authentication protocol for internet of drones. In2021 International Telecommunications Conference (ITC-Egypt) 2021 Jul 13 (pp. 1-4). IEEE.

[170] Sun G, Li C, Ma Y, Li S, Qiu J. End-to-end tcp congestion control as a classification problem. IEEE Transactions on Reliability. 2022 May 20;72(1):384-94.

[171] Aykurt K, Zerwas J, Blenk A, Kellerer W. When TCP Meets Reconfigurations: A Comprehensive Measurement Study. IEEE Transactions on Network and Service Management. 2023 Oct 25.

[172] Ramos D, Esparza O, Mata-Díaz J, Alins J. Evaluation of TCP Congestion Control Algorithms with traffic control policies in a PEP-based geosynchronous satellite scenario. Computer Networks. 2024 Feb 1;239:110131.

[173] Nikzad M, Jamshidi K, Bohlooli A, Faqiry FM. An accurate retransmission timeout estimator for content-centric networking based on the Jacobson algorithm. Digital Communications and Networks. 2022 Dec 1;8(6):1085-93.

[174] Luo C, Gu H, Zhu L, Zhang H. FlowStar: Fast Convergence Per-Flow State Accurate Congestion Control for InfiniBand. IEEE/ACM Transactions on Networking. 2024 Feb 13.

[175] Nyakomitta SP, Omollo V. Biometric-Based Authentication Model for E-Card Payment Technology. IOSR Journal of Computer Engineering (IOSRJCE). 2014;16(5):137-44.

[176] Dhamdhere A, Jiang H, Dovrolis C. Buffer sizing for congested internet links. InProceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies. 2005 Mar 13 (Vol. 2, pp. 1072-1083). IEEE.

[177] Anitha P, Vimala HS, Shreyas J. Comprehensive review on congestion detection, alleviation, and control for IoT networks. Journal of Network and Computer Applications. 2023 Oct 4:103749.

[178] Xie H, Li T. Revisiting loss recovery for high-speed transmission. In2022 IEEE Wireless Communications and Networking Conference (WCNC) 2022 Apr 10 (pp. 1987-1992). IEEE.

[179] Abubakar A, Oo KH. Window Size and Round-Trip-Time in a Network Transmission Session. In2018 International Conference on Information and Communication Technology for the Muslim World (ICT4M) 2018 Jul 23 (pp. 162-166). IEEE.

[180] Afanasyev A, Tilley N, Reiher P, Kleinrock L. Host-to-host congestion control for TCP. IEEE Communications surveys & tutorials. 2010 May 10;12(3):304-42.

[181] Xu Y, Shukla S, Guo Z, Liu S, Tam AS, Xi K, Chao HJ. RAPID: Avoiding TCP incast throughput collapse in public clouds with intelligent packet discarding. IEEE Journal on Selected areas in Communications. 2019 Jul 19;37(8):1911-23.

[182] Nyangaresi VO, Alsamhi SH. Towards secure traffic signaling in smart grids. In2021 3rd Global Power, Energy and Communication Conference (GPECOM) 2021 Oct 5 (pp. 196-201). IEEE.

[183] Zhou Z, Yang X. Speeding Up TCP with Selective Loss Prevention. In2021 IEEE 29th International Conference on Network Protocols (ICNP) 2021 Nov 1 (pp. 1-6). IEEE.

[184] Enachescu M, Ganjali Y, Goel A, McKeown N, Roughgarden T. Routers with Very Small Buffers. InINFOCOM 2006 Apr 23 (pp. 1-11).

[185] Chen X, Kim H, Aman JM, Chang W, Lee M, Rexford J. Measuring TCP round-trip time in the data plane. InProceedings of the Workshop on Secure Programmable Network Infrastructure 2020 Aug 14 (pp. 35-41).

[186] Karafillis P, Fouli K, ParandehGheibi A, Médard M. An algorithm for improving sliding window network coding in TCP. In2013 47th Annual Conference on Information Sciences and Systems (CISS) 2013 Mar 20 (pp. 1-5). IEEE.

[187] Shreedhar T, Panda R, Podanev S, Bajpai V. Evaluating QUIC performance over web, cloud storage, and video workloads. IEEE Transactions on Network and Service Management. 2021 Dec 10;19(2):1366-81.

[188] Kim GH, Cho YZ. Delay-aware BBR congestion control algorithm for RTT fairness improvement. IEEE Access. 2019 Dec 25;8:4099-109.

[189] Umran SM, Lu S, Abduljabbar ZA, Nyangaresi VO. Multi-chain blockchain based secure data-sharing framework for industrial IoTs smart devices in petroleum industry. Internet of Things. 2023 Dec 1;24:100969.

[190] Huang J, Qian F, Guo Y, Zhou Y, Xu Q, Mao ZM, Sen S, Spatscheck O. An in-depth study of LTE: Effect of network protocol and application behavior on performance. ACM SIGCOMM Computer Communication Review. 2013 Aug 27;43(4):363-74.

[191] Li L, Xu K, Wang D, Peng C, Zheng K, Mijumbi R, Xiao Q. A longitudinal measurement study of TCP performance and behavior in 3G/4G networks over high speed rails. IEEE/ACM transactions on networking. 2017 May 2;25(4):2195-208.

[192] Pravinbahi PR, Pravinchandra GA. TCP M-Start: A New Slow Start Method of TCP to Transfer Data Over Long Fat Pipe Network. International Journal of Intelligent Engineering & Systems. 2017 Jan 1;10(1).

[193] Vishwanath A, Sivaraman V, Thottan M. Perspectives on router buffer sizing: Recent results and open problems. ACM SIGCOMM Computer Communication Review. 2009 Mar 31;39(2):34-9.

[194] Chakravorty R, Katti S, Crowcroft J, Pratt I. Flow aggregation for enhanced TCP over wide-area wireless. InIEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No. 03CH37428) 2003 Mar 30 (Vol. 3, pp. 1754-1764). IEEE.

[195] Rasol M, Al Kasasbeh B, Al Adwan F. An improved secure SIP registration mechanism to avoid VoIP threats. International Journal of Cloud Applications and Computing (IJCAC). 2016 Apr 1;6(2):25-36.

[196] Sawabe A, Shinohara Y, Iwai T. Revisiting TCP Pacing for Throughput Performance Enhancement Over TDD Band in Private Mobile Networks. In2024 IEEE 21st Consumer Communications & Networking Conference (CCNC) 2024 Jan 6 (pp. 863-868). IEEE.

[197] Nyangaresi VO, Mohammad Z. Session Key Agreement Protocol for Secure D2D Communication. InThe Fifth International Conference on Safety and Security with IoT: SaSeIoT 2021 2022 Jun 12 (pp. 81-99). Cham: Springer International Publishing.

[198] Han J, Xue K, Li J, Xing Y, Yu R, Wei DS, Xue G. FMPTCP: Achieving High Bandwidth Utilization and Low Latency in Data Center Networks. IEEE Transactions on Communications. 2023 Oct 11.

[199] Michelinakis F, Kreitz G, Petrocco R, Zhang B, Widmer J. Passive mobile bandwidth classification using short lived TCP connections. In2015 8th IFIP Wireless and Mobile Networking Conference (WMNC) 2015 Oct 5 (pp. 104-111). IEEE.

[200] Mirkovic D, Armitage G, Branch P. A survey of round trip time prediction systems. IEEE Communications Surveys & Tutorials. 2018 Mar 19;20(3):1758-76.

[201] Kekely M, Kekely L, Kořenek J. General memory efficient packet matching FPGA architecture for future high-speed networks. Microprocessors and Microsystems. 2020 Mar 1;73:102950.

[202] Crichigno J, Csibi Z, Bou-Harb E, Ghani N. Impact of segment size and parallel streams on TCP BBR. In2018 41st International Conference on Telecommunications and Signal Processing (TSP) 2018 Jul 4 (pp. 1-5). IEEE.

[203] Liu S, Başar T, Srikant R. TCP-Illinois: A loss and delay-based congestion control algorithm for high-speed networks. InProceedings of the 1st international conference on Performance evaluation methodolgies and tools 2006 Oct 11 (pp. 55-es).

[204] Abduljabbar ZA, Omollo Nyangaresi V, Al Sibahee MA, Ghrabat MJ, Ma J, Qays Abduljaleel I, Aldarwish AJ. Session-Dependent Token-Based Payload Enciphering Scheme for Integrity Enhancements in Wireless Networks. Journal of Sensor and Actuator Networks. 2022 Sep 19;11(3):55.

[205] Kohler E, Handley M, Floyd S. Designing DCCP: Congestion control without reliability. ACM SIGCOMM Computer Communication Review. 2006 Aug 11;36(4):27-38.

[206] Cui C. Study on the Performance of TCP over 10Gbps High Speed Networks. Louisiana State University and Agricultural & Mechanical College; 2013.

[207] Arouche Nunes BA, Veenstra K, Ballenthin W, Lukin S, Obraczka K. A machine learning framework for TCP round-trip time estimation. EURASIP Journal on Wireless Communications and Networking. 2014 Dec;2014:1-22.

[208] Ahmad M, Hussain M, Abbas B, Aldabbas O, Jamil U, Ashraf R, Asadi S. End-to-end loss based TCP congestion control mechanism as a secured communication technology for smart healthcare enterprises. IEEE Access. 2018 Feb 19;6:11641-56.

[209] Gharakheili HH, Vishwanath A, Sivaraman V. Comparing edge and host traffic pacing in small buffer networks. Computer Networks. 2015 Feb 11;77:103-16.

[210] Alipio M, Tiglao NM, Bokhari F, Khalid S. TCP incast solutions in data center networks: A classification and survey. Journal of Network and Computer Applications. 2019 Nov 15;146:102421.

[211] Nyangaresi VO, Ma J. A Formally Verified Message Validation Protocol for Intelligent IoT E-Health Systems. In2022 IEEE World Conference on Applied Intelligence and Computing (AIC) 2022 Jun 17 (pp. 416-422). IEEE.

[212] Al Shinwan M, Abualigah L, Le ND, Kim C, Khasawneh AM. An intelligent long-lived TCP based on real-time traffic regulation. Multimedia Tools and Applications. 2021 May;80:16763-80.

[213] Jiang H, Wang Y, Lee K, Rhee I. DRWA: A receiver-centric solution to bufferbloat in cellular networks. IEEE Transactions on Mobile Computing. 2015 Dec 29;15(11):2719-34.

[214] Mehdizadeh A, Nagarajan M, Harun H, Mohammadpoor M. Congestion window scaling method to optimize delay in TCP/IP. Wireless Personal Communications. 2018 Aug;101(4):2227-39.

[215] Sarala S, Krishnamoorthi K. Enhanced packet routing queuing model in optical burst switching network using queue-based dynamic optical route scheduling. Microprocessors and Microsystems. 2020 Nov 1;79:103296.

[216] Al-Saadi R, Armitage G, But J, Branch P. A survey of delay-based and hybrid TCP congestion control algorithms. IEEE Communications Surveys & Tutorials. 2019 Mar 17;21(4):3609-38.

[217] Papadogiannakis A, Polychronakis M, Markatos EP. Stream-oriented network traffic capture and analysis for high-speed networks. IEEE Journal on Selected Areas in Communications. 2014 Sep 17;32(10):1849-63.

[218] Liu K, Lee JY. On improving TCP performance over mobile data networks. IEEE transactions on mobile computing. 2015 Nov 12;15(10):2522-36.

[219] Ahmad S, Arshad MJ. Enhancing fast TCP's performance using single TCP connection for parallel traffic flows to prevent head-of-line blocking. IEEE Access. 2019 Oct 9;7:148152-62.

[220] Abduljabbar ZA, Nyangaresi VO, Ma J, Al Sibahee MA, Khalefa MS, Honi DG. MAC-Based Symmetric Key Protocol for Secure Traffic Forwarding in Drones. InFuture Access Enablers for Ubiquitous and Intelligent Infrastructures: 6th EAI International Conference, FABULOUS 2022, Virtual Event, May 4, 2022, Proceedings 2022 Sep 18 (pp. 16-36). Cham: Springer International Publishing.

[221] Tao Y, Jiang J, Ma S, Wang L, Wang W, Li B. Unraveling the RTT-fairness Problem for BBR: A queueing model. In2018 IEEE global communications conference (GLOBECOM) 2018 Dec 9 (pp. 1-6). IEEE.

[222] Xue L, Kumar S, Cui C, Kondikoppa P, Chiu CH, Park SJ. Towards fair and low latency next generation high speed networks: AFCD queuing. Journal of Network and Computer Applications. 2016 Jul 1;70:183-93.

[223] Aoyagi S, Horie Y, Thi Thu Hien D, Duc Ngo T, Le DD, Nguyen K, Sekiya H. An Accurate Platform for Investigating TCP Performance in Wi-Fi Networks. Future Internet. 2023 Jul 19;15(7):246.

[224] Davydow A, Chuprikov P, Nikolenko SI, Kogan K. Competitive buffer management for packets with latency constraints. Computer Networks. 2021 Apr 22;189:107942.

[225] Louvros S, Paraskevas M, Chrysikos T. QoS-Aware Resource Management in 5G and 6G Cloud-Based Architectures with Priorities. Information. 2023 Mar 9;14(3):175.

[226] Nyangaresi VO. Provably Secure Pseudonyms based Authentication Protocol for Wearable Ubiquitous Computing Environment. In2022 International Conference on Inventive Computation Technologies (ICICT) 2022 Jul 20 (pp. 1-6). IEEE.

[227] Liu X, Ren F, Shu R, Zhang T, Dai T. Mitigating bufferbloat with receiver-based TCP flow control mechanism in cellular networks. InAdvances in Computer Communications and Networks From Green, Mobile, Pervasive Networking to Big Data Computing 2022 Sep 1 (pp. 65-90). River Publishers.

[228] Ramagundam S. Predicting broadband network performance with ai-driven analysis. Journal of Research Administration. 2023;5(2):11287-99.

[229] Hou Z, She C, Li Y, Quek TQ, Vucetic B. Burstiness-aware bandwidth reservation for ultra-reliable and low-latency communications in tactile Internet. IEEE Journal on Selected Areas in Communications. 2018 Oct 5;36(11):2401-10.

[230] Bouacida N, Shihada B. Practical and dynamic buffer sizing using LearnQueue. IEEE Transactions on Mobile Computing. 2018 Sep 2;18(8):1885-97.

[231] Kfoury EF, Gomez J, Crichigno J, Bou-Harb E. An emulation-based evaluation of TCP BBRv2 alpha for wired broadband. Computer Communications. 2020 Sep 1;161:212-24.

[232] Al Sibahee MA, Abduljabbar ZA, Luo C, Zhang J, Huang Y, Abduljaleel IQ, Ma J, Nyangaresi VO. Hiding scrambled text messages in speech signals using a lightweight hyperchaotic map and conditional LSB mechanism. Plos one. 2024 Jan 3;19(1):e0296469.

[233] Abdelmoniem AM, Bensaou B. Enhancing TCP via hysteresis switching: theoretical analysis and empirical evaluation. IEEE/ACM Transactions on Networking. 2023 Apr 7.

[234] Polese M, Chiariotti F, Bonetto E, Rigotto F, Zanella A, Zorzi M. A survey on recent advances in transport layer protocols. IEEE Communications Surveys & Tutorials. 2019 Aug 5;21(4):3584-608.

[235] Liu Q, Xu K, Wang H, Shen M, Li L, Xiao Q. Measurement, modeling, and analysis of TCP in high-speed mobility scenarios. In2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS) 2016 Jun 27 (pp. 629-638). IEEE.

[236] Mittal R, Lam VT, Dukkipati N, Blem E, Wassel H, Ghobadi M, Vahdat A, Wang Y, Wetherall D, Zats D. TIMELY: RTT-based congestion control for the datacenter. ACM SIGCOMM Computer Communication Review. 2015 Aug 17;45(4):537-50.

[237] Srinivasan S, Shippey S, Aryafar E, Chakareski J. FBDT: Forward and Backward Data Transmission Across RATs for High Quality Mobile 360-Degree Video VR Streaming. InProceedings of the 14th Conference on ACM Multimedia Systems 2023 Jun 7 (pp. 130-141).

[238] Nyangaresi VO. Lightweight anonymous authentication protocol for resource-constrained smart home devices based on elliptic curve cryptography. Journal of Systems Architecture. 2022 Dec 1;133:102763.

[239] Wang Z, Zeng X, Liu X, Xu M, Wen Y, Chen L. TCP congestion control algorithm for heterogeneous Internet. Journal of network and computer applications. 2016 Jun 1;68:56-64.

[240] MacDavid R, Chen X, Rexford J. Scalable real-time bandwidth fairness in switches. IEEE/ACM Transactions on Networking. 2023 Oct 9.

[241] Garetto M, Towsley D. An efficient technique to analyze the impact of bursty TCP traffic in wide-area networks. Performance Evaluation. 2008 Feb 1;65(2):181-202.

[242] Wang J, Yuan D, Luo W, Rao S, Sherratt RS, Hu J. Congestion control using in-network telemetry for lossless datacenters. Computers, Materials & Continua. 2023;75(1):1195-212.

[243] Varga B, Farkas J, Fejes F, Ansari J, Moldován I, Máté M. Robustness and Reliability Provided by Deterministic Packet Networks (TSN and DetNet). IEEE Transactions on Network and Service Management. 2023 Jun 9.

[244] Zhang H, Ma J, Qiu Z, Yao J, Sibahee MA, Abduljabbar ZA, Nyangaresi VO. Multi-GPU Parallel Pipeline Rendering with Splitting Frame. InComputer Graphics International Conference 2023 Aug 28 (pp. 223-235). Cham: Springer Nature Switzerland.

[245] Clayman S, Sayıt M. Low latency low loss media delivery utilizing in-network packet wash. Journal of Network and Systems Management. 2023 Jan;31(1):29.

[246] Rojas-Cessa R, Kaymak Y, Dong Z. Schemes for fast transmission of flows in data center networks. IEEE Communications Surveys & Tutorials. 2015 Apr 28;17(3):1391-422.

[247] Najmuddin S, Asim M, Munir K, Baker T, Guo Z, Ranjan R. A BBR-based congestion control for delay-sensitive real-time applications. Computing. 2020 Dec;102:2541-63.

[248] Escudero-Sahuquillo J, Gran EG, Garcia PJ, Flich J, Skeie T, Lysne O, Quiles FJ, Duato J. Efficient and cost-effective hybrid congestion control for HPC interconnection networks. IEEE transactions on parallel and distributed systems. 2014 Feb 24;26(1):107-19.

[249] Stephens B, Cox AL, Singla A, Carter J, Dixon C, Felter W. Practical DCB for improved data center networks. InIEEE INFOCOM 2014-IEEE Conference on Computer Communications 2014 Apr 27 (pp. 1824-1832). IEEE.

[250] Nyangaresi VO. Lightweight key agreement and authentication protocol for smart homes. In2021 IEEE AFRICON 2021 Sep 13 (pp. 1-6). IEEE.

[251] Lee JY, Kim BC, Kwon Y, Han K. Coupled CUBIC Congestion Control for MPTCP in Broadband Networks. Computer Systems Science & Engineering. 2023 Apr 1;45(1).

[252] Zingirian N. Multi-Stream TCP Design. In2023 IEEE 19th International Conference on Intelligent Computer Communication and Processing (ICCP) 2023 Oct 26 (pp. 123-130). IEEE.

[253] Ahmad SZ, Khalid S. Optimizing Data Transport Efficiency in Datacenters through Traffic Shaping of BBR Congestion Control. J. Commun.. 2023 Feb;18(2):97-108.

[254] Wang X, He X, Ren H. Advanced FEC for 200 Gb/s Transceiver in 800 GbE and 1.6 TbE Standard. IEEE Communications Standards Magazine. 2023 Sep;7(3):56-62.

[255] Li T, Zheng K, Xu K, Jadhav RA, Xiong T, Winstein K, Tan K. Tack: Improving wireless transport performance by taming acknowledgments. InProceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication 2020 Jul 30 (pp. 15-30).

[256] Abdullah S. Enhancing the TCP Newreno Fast RecoveryAlgorithm on 5G Networks. Journal of Computing and Communication. 2024 Jan 31;3(1):33-43.

[257] Abdelsalam A, Luglio M, Patriciello N, Roseti C, Zampognaro F. TCP Wave over Linux: a disruptive alternative to the traditional TCP window approach. Computer Networks. 2021 Jan 15;184:107633.

[258] Qiu Z, Ma J, Zhang H, Al Sibahee MA, Abduljabbar ZA, Nyangaresi VO. Concurrent pipeline rendering scheme based on GPU multi-queue and partitioning images. InInternational Conference on Optics and Machine Vision (ICOMV 2023) 2023 Apr 14 (Vol. 12634, pp. 143-149). SPIE.

[259] Zhao Z, Cao W. Improved Bottleneck Bandwidth and Round-Trip Propagation Congestion Control Algorithm for Round-Trip Time Fairness. Journal of Advanced Computational Intelligence and Intelligent Informatics. 2023 May 20;27(3):346-51.

[260] Huh EN, Choo H. Performance enhancement of TCP in high-speed networks. Information Sciences. 2008 Jan 15;178(2):352-62.

[261] Abed GA, Ismail M, Jumari K. Exploration and evaluation of traditional TCP congestion control techniques. Journal of King Saud University-Computer and Information Sciences. 2012 Jul 1;24(2):145-55.

[262] Nandhini C, Gupta GP. Exploration and Evaluation of Congestion Control Algorithms for Data Center Networks. SN Computer Science. 2023 Jun 30;4(5):509.

[263] Scholz D, Jaeger B, Schwaighofer L, Raumer D, Geyer F, Carle G. Towards a deeper understanding of TCP BBR congestion control. In2018 IFIP networking conference (IFIP networking) and workshops 2018 May 14 (pp. 1-9). IEEE.

[264] Nyangaresi VO. Terminal independent security token derivation scheme for ultra-dense IoT networks. Array. 2022 Sep 1;15:100210.

[265] Lu Y, Cui C, Ma X, Ruan Z. A3DCT: A cubic acceleration TCP for data center networks. Journal of Network and Computer Applications. 2023 Jul 1;216:103654.

[266] Lu Y, Cui C, Ma X, Ruan Z. A3DCT: A cubic acceleration TCP for data center networks. Journal of Network and Computer Applications. 2023 Jul 1;216:103654.

[267] Jamali S, Alipasandi N, Alipasandi B. TCP pegas: A PSO-based improvement over TCP vegas. Applied Soft Computing. 2015 Jul 1;32:164-74.

[268] de Almeida LC, da Silva WR, Tavares TC, Pasquini R, Papagianni C, Verdi FL. DESiRED-Dynamic, Enhanced, and Smart iRED: A P4-AQM with Deep Reinforcement Learning and In-band Network Telemetry. Computer Networks. 2024 Mar 16:110326.

[269] Beshley M, Kryvinska N, Seliuchenko M, Beshley H, Shakshuki EM, Yasar AU. End-to-End QoS “smart queue” management algorithms and traffic prioritization mechanisms for narrow-band internet of things services in 4G/5G networks. Sensors. 2020 Apr 19;20(8):2324.

[270] Sup LM, de Moraes RM, Bauchspiess A. Explicit non-congestion notification: A new AQM approach for TCP networks. In2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC) 2017 Jun 26 (pp. 1239-1244). IEEE.

[271] Kumar S, Chinthaginjala R, Anbazhagan R, Nyangaresi VO, Pau G, Varma PS. Submarine Acoustic Target Strength Modelling at High-Frequency Asymptotic Scattering. IEEE Access. 2024 Jan 1.

[272] Weber D, Fuchs C, Auler N, Schütz B, Aschenbruck N. Multipath TCP Scheduling in the Age of Buffer Bloat Optimizations. In2023 IEEE 48th Conference on Local Computer Networks (LCN) 2023 Oct 2 (pp. 1-8). IEEE.

[273] Karpowicz MP. Adaptive tuning of network traffic policing mechanisms for DDoS attack mitigation systems. European Journal of Control. 2021 Sep 1;61:101-18.

[274] Alrshah MA, Al-Maqri MA, Othman M. Elastic-TCP: Flexible congestion control algorithm to adapt for high-BDP networks. IEEE Systems Journal. 2019 Feb 14;13(2):1336-46.

[275] Priyanka D, Krishna YS. Analysis of Transmission Control Protocol in Next Generation Networks. I-Manager's Journal on Wireless Communication Networks. 2023 Jan 1;11(2).

[276] Xie Y, Jiang X, Gong G, Jiang Z, Jin G, Chen H. Yinker: A flexible BBR to achieve the high-throughput and low-latency data transmission over Wi-Fi and 5G networks. Computer Networks. 2023 Feb 1;222:109530.

[277] Nyangaresi VO. A Formally Validated Authentication Algorithm for Secure Message Forwarding in Smart Home Networks. SN Computer Science. 2022 Jul 9;3(5):364.

[278] Soldatos J, Vayias E, Kormentzas G. On the building blocks of quality of service in heterogeneous IP networks. IEEE Communications Surveys & Tutorials. 2005 May 2;7(1):69-88.

[279] Al-Dweik A, Iraqi Y, Mukhtar H, Naeem M, Hossain E. Hybrid Automatic Repeat Request (HARQ) in Wireless Communications Systems and Standards: A Contemporary Survey. Authorea Preprints. 2023 Oct 30.

[280] Esposito C, Bruno A, Cattaneo G, Palmieri F. On the optimal tuning and placement of FEC codecs within multicasting trees for resilient publish/subscribe services in edge-IoT architectures. Future generation computer systems. 2018 Nov 1;88:140-50.

[281] Lin J, Cui L, Zhang Y, Tso FP, Guan Q. Extensive evaluation on the performance and behaviour of TCP congestion control protocols under varied network scenarios. Computer Networks. 2019 Nov 9;163:106872.

[282] Mishra A, Sun X, Jain A, Pande S, Joshi R, Leong B. The great internet TCP congestion control census. Proceedings of the ACM on Measurement and Analysis of Computing Systems. 2019 Dec 17;3(3):1-24.

[283] Prados-Garzon J, Taleb T. Asynchronous time-sensitive networking for 5G backhauling. IEEE Network. 2021 Mar 8;35(2):144-51.

[284] Al-Maqri MA, Alrshah MA, Othman M. Review on QoS provisioning approaches for supporting video traffic in IEEE802. 11e: Challenges and issues. IEEE Access. 2018 Sep 28;6:55202-19.