World Journal of Advanced Engineering Technology and Sciences

eISSN: 2582-8266

Cross Ref DOI: 10.30574/wjaets

Journal homepage: https://wjaets.com/

(Review Article)

TCP/IP stack transport layer performance, privacy, and security issues

Oroo Oyondi Felix *

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

World Journal of Advanced Engineering Technology and Sciences, 2024, 11(02), 175–200

Publication history: Received on 01 February 2024; revised on 13 March 2024; accepted on 15 March 2024

Article DOI: https://doi.org/10.30574/wjaets.2024.11.2.0098

Abstract

Transmission Control Protocol/ Internet Protocol (TCP/IP) is the backbone of Internet transmission. The Transport Layer of the TCP/IP stack, which includes TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) protocols, plays a crucial role in ensuring reliable communication between devices over a network. To come up with measures that make networks more secure, it is important to learn about the vulnerabilities that exist in the transport TCP/IP stack and then have an understanding of the typical attacks carried out in such layer. This paper explores how the TCP Protocol works, the TCP/IP 3 Way Handshake, TCP Header Structure, the typical vulnerabilities and the classical attacks of transport layer TCP/IP, tools, and solutions adopted to prevent and reduce the chances of some of these attacks. The findings indicated that the major TCP/ IP stack transport layer threats include Finger printing, SYN Flood, TCP reassembly and sequencing, IP Spoofing, TCP session hijacking, RST and FIN denial of service attack, Ping of Death, Low Rate/ Shrew Attacks. Their preventive measures and mechanisms are discussed.

Key Words: TCP; TLS; Encryption; TCP Header; SYN; ACK.

1. Introduction

The Transport Layer is responsible for end-to-end communication between hosts on a network. It includes protocols like TCP and UDP, which provide different levels of reliability and performance [1]-[4]. TCP is connection-oriented, provides reliable, ordered delivery of data, while UDP is connectionless, and provides a best-effort delivery mechanism. TCP includes both a flow control mechanism, error checking and congestion control mechanism. Flow control means that the receiver's TCP is able to control the size of the segment dispatched by the sender's TCP [5] [6]. The receiver's TCP accomplishes by putting to use the Window field of an acknowledgment packet. Congestion control means that the sender's TCP varies the rate at which it places the packets on the wire based on the traffic congestion on the route between the sender and the receiver. The sender TCP can measure traffic congestion through either the non-arrival of an expected ACK packet or by the arrival of three identical ACK packets consecutively The differences in levels of TCP reliability [7] have implications for performance, privacy, and security [8], [9]. At the Transport Layer of the TCP/IP stack, there are several important considerations regarding performance, privacy, and security and the attacks.

1.1. Performance

TCP performance is a critical aspect of network communication, influencing the efficiency, reliability, and responsiveness of data transfer. TCP achieves reliability through mechanisms like error detection, acknowledgment, and retransmission of lost packets, ensuring data integrity even in the face of network congestion or packet loss [10]-[13]. However, these mechanisms can introduce overhead and latency, impacting performance, particularly in high-latency or high-loss network environments. To mitigate these issues, various TCP optimization techniques such as window scaling, selective acknowledgment, and congestion control algorithms like TCP Vegas or TCP Cubic are employed to adapt TCP's behavior dynamically to network conditions, optimizing throughput and minimizing latency.

* Corresponding author: Oroo Oyondi Felix.

Balancing reliability with performance remains a constant challenge in TCP design, as improving one aspect often comes at the expense of another, necessitating continuous refinement and adaptation to meet the evolving demands of modern network applications. Figure 1 shows the TCP/IP protocol suite encapsulation model.

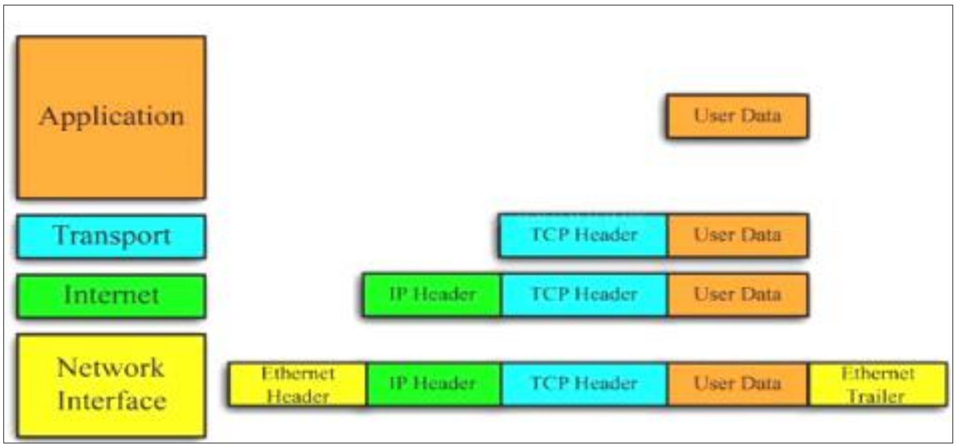

Figure 1 TCP/IP Encapsulation model

The TCP/IP transport layer protocols, primarily Transmission Control Protocol (TCP) and User Datagram Protocol (UDP), play pivotal roles in facilitating reliable and efficient communication across networks. TCP is a connection-oriented protocol that ensures reliable data delivery through features such as flow control, error detection, and retransmission of lost packets [14], [15]. It establishes a virtual connection between sender and receiver, guaranteeing that data is delivered in the correct order and without errors. TCP achieves this reliability by employing mechanisms like sequence numbers, acknowledgment messages, and sliding window flow control, making it well-suited for applications that prioritize data integrity and completeness, such as file transfer, email, and web browsing.

In contrast, UDP is a connectionless protocol that provides a lightweight and fast transmission mechanism with minimal overhead. UDP sacrifices reliability for speed, as it does not implement features like acknowledgment or error recovery. Instead, UDP simply encapsulates data into datagrams and sends them across the network without establishing a connection or ensuring delivery [16], [17]. This makes UDP ideal for applications that prioritize speed and efficiency over reliability, such as real-time multimedia streaming, online gaming, and VoIP (Voice over Internet Protocol). While UDP lacks the built-in mechanisms for reliability found in TCP, it allows for faster transmission of time-sensitive data, making it a valuable tool in a variety of network applications.

1.2. Privacy

Privacy within the TCP/IP suite, which encompasses various protocols facilitating internet communication, is a multifaceted issue influenced by several factors. At the transport layer, TCP and UDP protocols themselves do not inherently prioritize privacy; rather, they primarily focus on reliable data delivery and efficient transmission [17], [18]. However, privacy concerns often arise at higher layers of the protocol stack, such as the application layer, where sensitive user data is transmitted over the network [20]. Encryption protocols like TLS (Transport Layer Security) can be employed to secure communication channels, ensuring privacy by encrypting data transmitted between endpoints. By implementing end-to-end encryption, TLS protects data from interception and eavesdropping, thus safeguarding user privacy in transit.

Furthermore, privacy in the TCP/IP suite is influenced by the design and implementation of various network applications and services. For instance, web browsers, email clients, and messaging applications handle user data differently, and their privacy practices vary widely [21], [22]. Some applications may collect and transmit user data without adequate encryption or consent, raising privacy concerns. Additionally, network protocols like DNS (Domain Name System) and DHCP (Dynamic Host Configuration Protocol) within the TCP/IP suite can inadvertently expose user information, as they often transmit data in plaintext, leaving it vulnerable to interception [23]-[26]. Addressing privacy concerns at the application and protocol level requires careful consideration of data handling practices, implementation of encryption, and adherence to privacy regulations and standards.

Moreover, the proliferation of IoT (Internet of Things) devices and the integration of TCP/IP protocols into various smart devices introduce new privacy challenges. These devices often collect and transmit sensitive user data, including personal information and behavioral patterns, raising concerns about data privacy and security [27], [28]. With the growing interconnectedness of devices and the internet, ensuring privacy within the TCP/IP suite necessitates comprehensive privacy-by-design principles, robust encryption mechanisms, and transparent data handling practices. Additionally, regulatory frameworks like GDPR (General Data Protection Regulation) and CCPA (California Consumer Privacy Act) play crucial roles in shaping privacy standards and holding organizations accountable for protecting user data across the TCP/IP ecosystem [29]-[32].

1.3. Security

Security issues within the TCP/IP suite represent a significant challenge due to the vast array of protocols and layers involved in internet communication. At the network layer, IP (Internet Protocol) is inherently vulnerable to various attacks such as IP spoofing, where attackers forge the source IP address of packets to impersonate legitimate users or bypass access controls [33]-[38]. Additionally, IP fragmentation attacks exploit the fragmentation and reassembly process of IP packets to evade detection and overwhelm network resources. These vulnerabilities highlight the importance of implementing security measures like packet filtering, ingress and egress filtering, and network segmentation to mitigate the risk of network-layer attacks and protect against unauthorized access.

Moreover, at the transport layer, TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) present security challenges related to session hijacking, packet sniffing, and denial-of-service (DoS) attacks. TCP-based attacks, such as SYN flooding, exploit the three-way handshake process to exhaust server resources and disrupt network services [39]-[44]. UDP-based attacks, on the other hand, leverage the connectionless nature of UDP to flood target systems with a high volume of malicious traffic, causing network congestion and service outages. To address these vulnerabilities, network administrators can implement techniques like TCP SYN cookies, rate limiting, and stateful inspection firewalls to detect and mitigate transport-layer attacks, ensuring the integrity and availability of network services [45].

Furthermore, security issues within the TCP/IP suite extend to the application layer, where protocols like HTTP, SMTP, and FTP are vulnerable to various attacks such as cross-site scripting (XSS), SQL injection, and email spoofing [46], [47]. These attacks exploit vulnerabilities in web applications, email servers, and file transfer mechanisms to compromise user data, exfiltrate sensitive information, or disrupt service availability. Additionally, insecure authentication mechanisms and insufficient encryption protocols within application-layer protocols expose user credentials and sensitive data to interception and unauthorized access [48]-[50]. To enhance security at the application layer, organizations can implement secure coding practices, deploy web application firewalls (WAFs), and enforce encryption standards like HTTPS and SFTP to protect against common attacks and safeguard user privacy.

Moreover, the proliferation of IoT (Internet of Things) devices and the integration of TCP/IP protocols into various smart devices introduce new security challenges, including device hijacking, botnet attacks, and data breaches. Insecure default configurations, lack of firmware updates, and insufficient authentication mechanisms in IoT devices expose them to exploitation by malicious actors, leading to widespread vulnerabilities and potential compromises of network infrastructure [51], [52]. Addressing security issues within the TCP/IP suite requires a holistic approach encompassing network monitoring, threat intelligence, vulnerability management, and security awareness training to detect, prevent, and mitigate security breaches across all layers of the internet protocol stack [53]-[56]. Additionally, collaboration between industry stakeholders, government agencies, and standards bodies is essential to develop and enforce security best practices and regulatory frameworks to protect against evolving cyber threats in an increasingly interconnected world [57], [58].

2. TCP 3 Way Handshake Protocol

TCP needs three handshakes to establish the connection, as shown in Figure 2. Multiple TCP socket connections can be transmitted in both directions simultaneously [59]. A three-way handshake is also known as a TCP handshake or SYN-SYN-ACK, and requires both the client and server to exchange SYN (synchronization) and ACK (acknowledgment) packets before actual data communication begins [60], [61]. In Step one (SYN), the client sends a SYN message. The client wants to establish a connection with a server, so it sends a segment with SYN (Synchronize Sequence Number) which informs the server that the client is likely to start communication and with what sequence number it starts segments [62]-[64]. In **Step two (SYN+ACK)** [65] the server replies with an **SYN/ACK** message. **SYN-ACK** signal bits are set. In **Step three**, **ACK**nowledgement (**ACK**) signifies the response of the segment it received and SYN signifies with what sequence number it is likely to start the segments with [66], which then responds with an ACK message [67]. In this final part, the client acknowledges the response of the server and they both establish a reliable connection [68] with which they will start the actual data transfer.

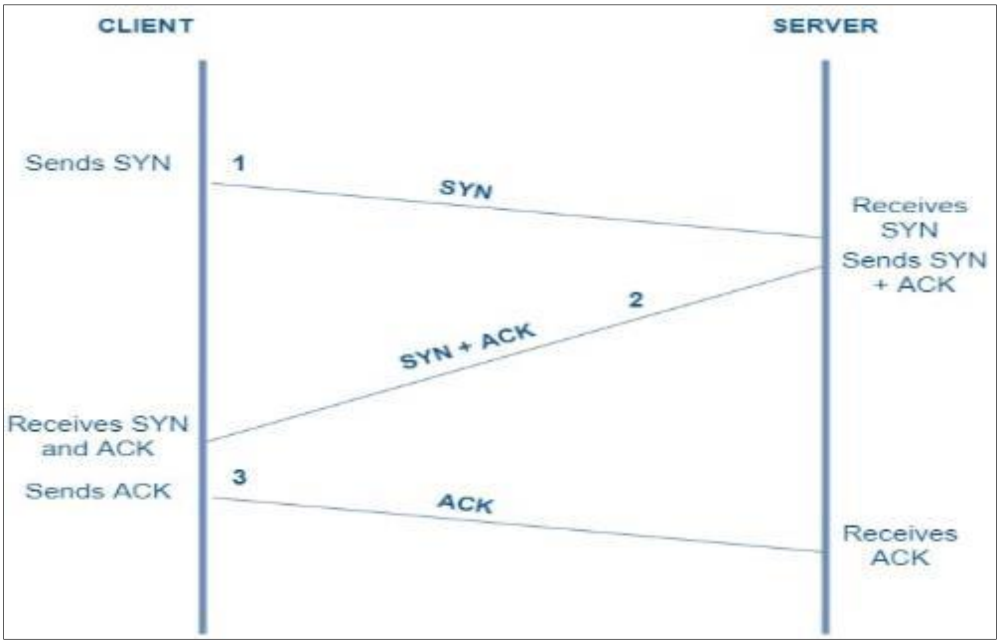

Figure 2 Three-Way TCP Handshake

Generally, the three messages transmitted by TCP to negotiate and start a TCP session are nicknamed SYN, *SYN-ACK*, and ACK for SYNchronize, SYNchronize-ACKnowledgement, and ACKnowledge respectively [69], [70]. The three-message mechanism enables the transport layer to pass information back and forth to two communicating computers to negotiate the parameters of the connection before transmitting data. This handshake step happens after a DNS lookup and before the TLS handshake, when creating a secure connection [71]. Each side of the connection via a four-way handshake can terminate the connection independently after an error occurs in the communication [72], [73].

2.1. TCP Header Structure

The TCP header structure consists of several fields that govern the behavior and characteristics of TCP segments. As shown in Figure 3, these fields include the source and destination port numbers, which identify the endpoints of the communication; sequence and acknowledgment numbers, used for reliable data delivery [74] and flow control; TCP flags such as SYN, ACK, FIN, and RST, which manage connection establishment, acknowledgment, and termination; window size, indicating the amount of data that can be sent without acknowledgment; checksum, providing error detection for the TCP header and data; and urgent pointer, used to indicate urgent data within the segment [75], [76]. Each field within the TCP header serves a specific purpose in facilitating reliable, connection-oriented communication between hosts, enabling features such as sequencing, acknowledgment, flow control, and error detection to ensure efficient and robust data transmission over IP networks.

In TCP, flags indicate a particular connection state or handle control of a specific connection [77], [78]. Flags are also called control bits. Each flag corresponds to 1-bit information. The most commonly used flags are **SYN**, **URG**, **ACK**, **PSH**, **FIN**, and **RST**. TCP uses a variable-length header to support data transmissions. TCP Header is larger at 20 bytes with an option for additional data [79]. The header can have anywhere between 20 and 60 bytes [80], [81].

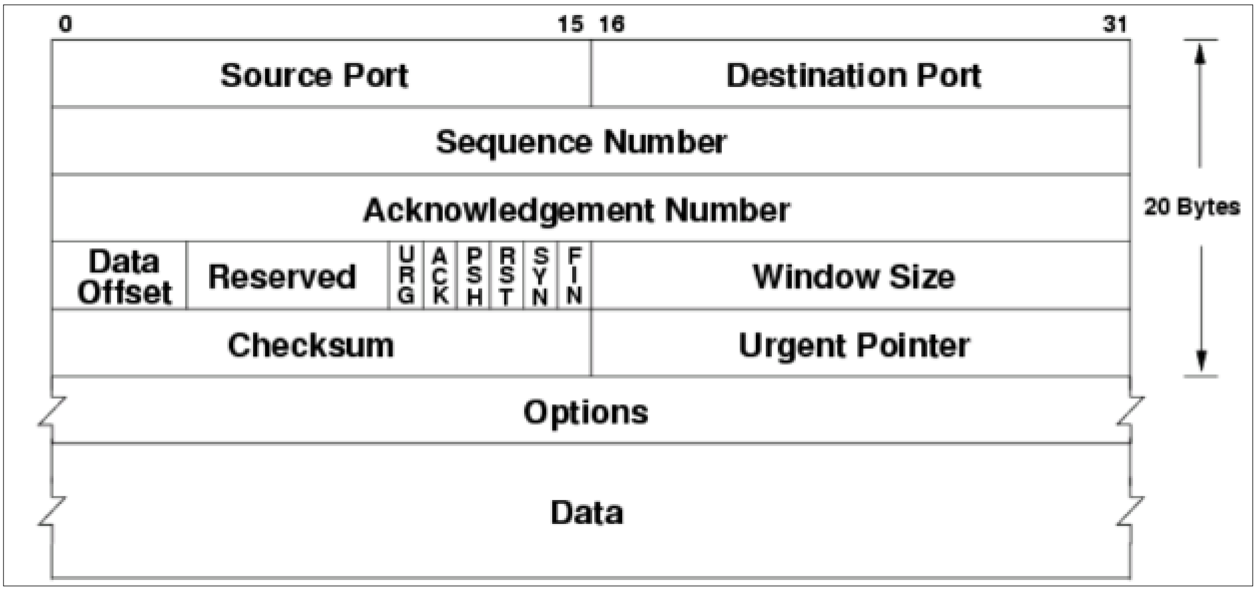

Figure 3 TCP Header Structure

The Source Port is a 16-bit field that indicates the port number of the sending device where the data originates. It is a randomly assigned a field [82].

In Destination Port is the field indicates the port number on the receiving device where the data should be delivered. It is 16 bits field [83].

In Sequence Number part, TCP converts data into bytes and the collection of bytes is known as segment. Each TCP segment is assigned a sequence number, which helps the receiving end to reassemble the data in the correct order. It is a 32-bit value [84].

In TCP Acknowledgment Number, the data transmission is acknowledged to ensure reliability [85], [86]. This field contains the sequence number that the receiving device expects to receive next. Acknowledgment no is always an incremental value i.e., if the sequence number is x, than Acknowledgment no is set to **x+1**.

At Data Offset field determines the size of the TCP header. It is necessary to locate the start of the data payload. It is a 4 bits field [86].

In Reserved field bits are reserved and set to zero [87].

The Control Bits (Flags), also called flags or TCP flags, are used to control and manage aspects of TCP connection and data transmission. Some common flags include are described in Table 1 that follows.

Table 1 TCP Header flags

Flag	**Description**
URG (Urgent)	This bit can be 0 or 1. When this bit is 1, it implies that the data should be treated as a priority. For example, data is always sent in a seq. but we have some urgent data bits that should be sent first. In that case, the Urgent bit is set ON for that particular data, and that data is sent first [88], [89].
ACK (Acknowledgment)	Indicates whether the acknowledgment number field is valid or not. If ACK is 1 it implies that the acknowledgment number is valid and if ACK is 0, it means that the segment is missing acknowledgment [90].
PSH (Push)	In general, applications collect a certain number of data and then process it. When the Push flag is set ON, it tells the application to transmit the data immediately and not wait for data to stack to fill the entire TCP segment [91].

RST (Reset)	Resets the connection. If it is set to 1, the connection is abruptly reset.
SYN (Synchronize)	Initiates a connection and synchronizes [92] sequence numbers. It is used in the 3-way handshake process [93].
FIN (Finish)	The fin flag is used to terminate the TCP connection [94]-[96]. Whenever Host wants to end the connection with the receiving end, it sends data with FIN flag 1. Since TCP works in a full duplex mode, receiving end should also set its FIN flag as 1.
Window Size	This field indicates the size of the receiving device's window, which helps in flow control. It is a 16-bit field. It is used for flow control between the sender and receiver [97].
Checksum	This is a 16-bit field numerical value calculated from the TCP header and data payload to detect errors during transmission. TCP header checksum option improves performance [98] over lossy links [99].
Urgent Pointer (URG)	This flag is set, and points to the last urgent data byte in the TCP segment i.e., it tells about the sequence number of the last urgent data byte. It is a 16-bit field [100].
Optional filed	This flag contains additional parameters or information related to the TCP connection [101], [102].

3. TCP Congestion Control

Congestion Control is a mechanism that controls the entry of data packets into a transport protocol, enabling a better use of a shared infrastructure and avoiding congestive collapse. Transport layer is the right layer to implement congestion control since it resides between application layer and network layer [103], [104]. There are three ways to deal with congestion, depending on the Quality of Service (QoS) requirements for each session. By default, overflow packets are discarded without informing the sender. Figure 4 shows the TCP congestion control mechanism.

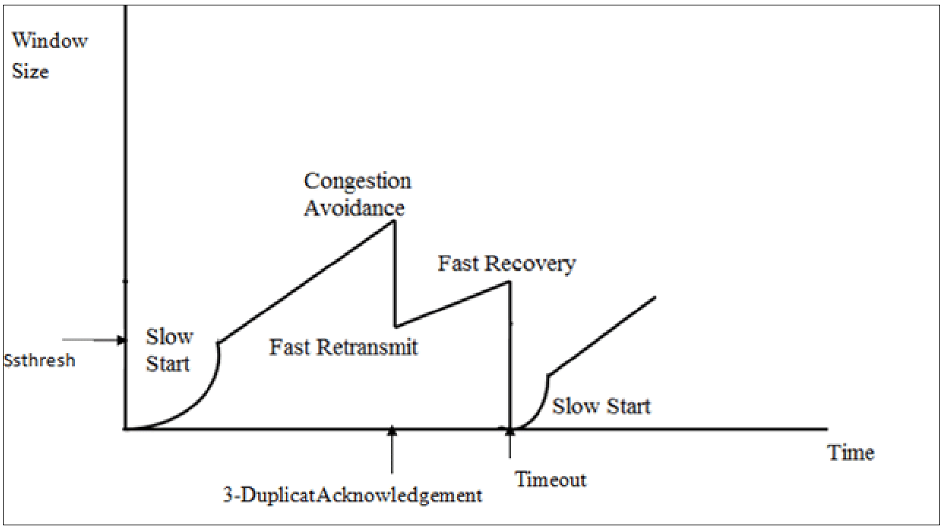

Figure 4 TCP congestion Control

Since TCP must guarantee reliability [105] in communications, it re-transmits a TCP segment when an ACK is not received in a certain period or when three duplicate ACKs are received consecutively (a condition triggered by the arrival of an out-of-order segment at the receiver; the duplicate ACK being for the last in-order segment received). Figure 5 shows the TCP congestion control components. As to how frequently a TCP segment is retransmitted is based on what is known as a Congestion Avoidance Algorithm [106]. TCP Congestion Avoidance Algorithm has a good overall summary of the different versions [107]-[109].

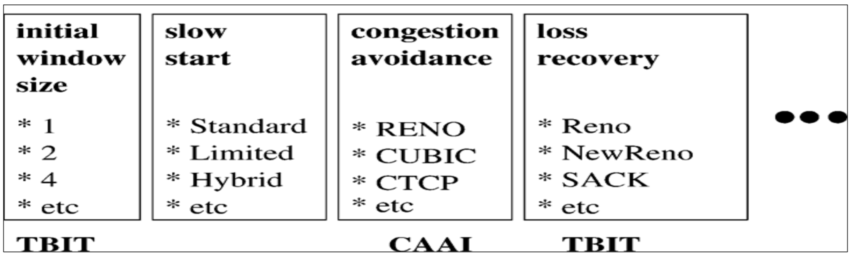

Figure 5 TCP congestion control components

The TCP congestion control algorithm has three major phases: The Low start, Congestion avoidance and Congestion detection and fast recovery. Traffic dynamics in the Internet are heavily influenced by the behavior of the TCP Congestion Avoidance algorithm. TCP congestion control affects the round-trip time (RTT) of packets within the flow (i.e., the flow RTT): an endpoint sends packets at higher throughput, increasing the occupancy of the bottleneck [110] buffer, thereby increasing the RTT of packets in the flow.

4. TCP Vegas, Tahoe/Reno and Cubic performance in Congestion Control Avoidance

TCP Vegas enhances the congestion avoidance control algorithm of TCP Reno. In this case, TCP Vegas dynamically increases or decreases its sending window size according to observed RTTs (Round Trip Times) of sending packets, and therefore, TCP Vegas does not suffer from packet retransmissions [111],[112]. TCP Tahoe/Reno is a classic congestion control algorithm that uses a mechanism called Additive Increase Multiplicative Decrease (**AIMD**) to adjust the TCP window size, which is the amount of data that can be sent without waiting for an Acknowledgement [113], [114]. It employs a linear function. It increases the window size by one segment for every Round Trip Time (RTT) until a packet loss is detected which indicates a congestion. Then, it halves the window size and enters a fast recovery phase, where it increases the window size by one segment for every duplicate acknowledgement (ACK) received. This way, TCP Reno tries to maintain a high throughput while avoiding congestion collapse. TCP Cubic uses a cubic function. After packet loss, Reno halves the window size whereas Cubic reduces it by a smaller factor [115]. TCP Cubic is more aggressive than TCP Reno in increasing the window size after a packet loss but also more conservative in reducing it. TCP Cubic also adapts to different network environments, such as high- bandwidth high- delay networks (HBHD), by using a scaling factor that depends on RTT. It aims to achieve a fair and efficient allocation of bandwidth while minimizing packet loss and delay.

Apart from TCP, there are other congestion control algorithms such as Explicit Congestion Notification (ECN), Stream Congestion Transmission Protocol (SCTP), and Data Center TCP (DCTCP), each designed to address specific network scenarios and requirements.

4.1. Slow Start Operation

A sender attempts to communicate to a receiver. The sender's initial packet contains a small congestion window, which is determined based on the sender's maximum window [116], [117]. The receiver acknowledges the packet and responds with its own window size. If the receiver fails to respond, the sender knows not to continue sending data. After receiving the acknowledgement, the sender increases the next packet's window size. The window size gradually increases until the receiver can no longer acknowledge each packet, or until either the sender or the receiver's window limit is reached. Once a limit has been determined, slow start's job is done. Other congestion control algorithms take over to maintain the speed of the connection.

4.2. Challenges in identifying the type of congestion

The server point of view has several advantages, the most important being that it has direct information about outgoing packets and TCP state [118]-[120]. However, even with a detailed view of the flow, distinguishing between the two types of congestion that is listed above is challenging. Some techniques include analyzing the flow throughput, TCP states, and/or flow packet arrivals or Round Trip Time (RTT). Each has its advantages and drawbacks. Information about flow throughput [121] is insufficient to determine the type of congestion unless we also know the actual service plan of the client. TCP state analysis can help us analyze TCP state transitions and flow behavior; however, it does not help us differentiate between different kinds of congestion. Transitions to/from the fast retransmit or the retransmission timeout state can potentially tell us about congestion events. However, it is difficult to parameterize and model these state changes. Simple techniques such as modeling the total number of fast retransmit and timeout states per time interval or the time to the first retransmit state have the same difficulty that it varies according to the path latency, service plan of the client, loss-rate, and cross-traffic, which are difficult to account for in controlled TCP settings. Packet arrival patterns are used to uncover a congested path [122]. Such techniques typically has the requirement that it be downstream of the point of congestion to be able to measure packet arrival rate. This is not possible with the server point of view, nor from the packet sender, unless they have access to network packets. Though packet spacing can be approximated by analyzing ACK arrival patterns, ACKs can be noisy, and cannot tell us any more than that the flow encountered congestion. Flow RTT, contains information about the condition of the underlying path [123]. In particular, the RTTs of packets in a flow allow one to distinguish between an empty bottleneck buffer (increasing RTT as the flow fills up the buffer) and a busy buffer (RTT is relatively stable as it is dominated by the added latency due to an already full buffer). Flow RTTs are useful only during the slow start period, but fortunately, this short interval is sufficient for one to be able to distinguish the two congestion states.

5. Vulnerabilities and threats at transport layer and their counter measures

A vulnerability is a flaw or weakness in an asset's design, implementation, or operation and management that could be exploited by a threat [124]-[129]. A threat is a potential for a threat agent to exploit a vulnerability. A risk is the potential for loss when the threat happens Insufficient Transport Layer Protection is a security weakness caused by applications not taking any measures to protect network traffic. During authentication, applications may use SSL/TLS, but they often fail to make use of it elsewhere in the application, thereby leaving data and session IDs exposed. Discussed below are the attacks at the transport layer.

5.1. Finger printing a system

Fingerprinting is used to discover open ports and services that are running open on the target system. From a hacker's point of view, fingerprinting is done before the exploitation phase, as the more information a hacker can obtain about a target, the hacker can then narrow its attack scope and use specific tools to increase the chances of successfully compromising the target machine [127], [128]. Figure 6 illustrates an operating system fingerprinting process. The most complete and widely used TCP/IP fingerprinting tool today is nmap. It uses a database of over 450 fingerprints to match TCP/IP stacks to a specific operating system or hardware platform. This database includes routers, switches, firewalls, and many other systems.

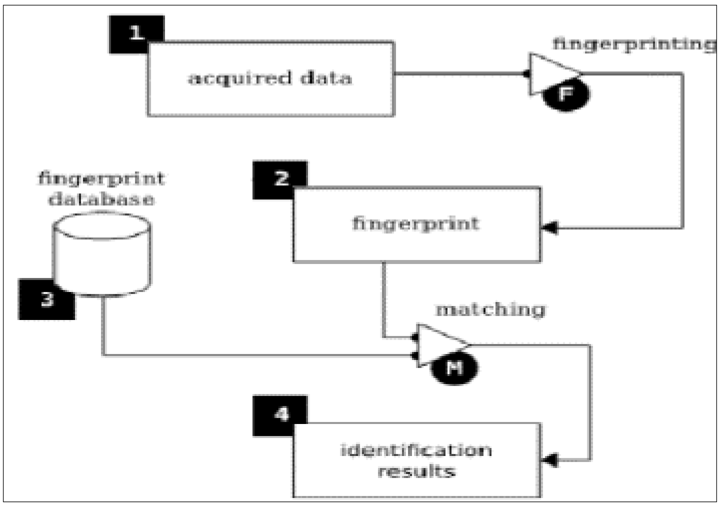

Figure 6 Operating system fingerprinting process

Any system that speaks TCP/IP is potentially in the database, which is updated frequently. Nmap fingerprints a system in three steps. First, it performs a port scan to determine a set of open and closed TCP and UDP ports [130]-[134]. Second, it generates specially formed packets, sends them to the remote host, and listens for responses. Third, it uses the results from the tests to determine a matching entry in its database of fingerprints. For example, we have a target machine 192.168.171.25, on a network. As a hacker would like to know which TCP ports are open, the services that use the open ports, and the service daemon(a service responsible for starting standard Internet services [135] when a system boots, they use transfer control protocol-TCP, Stream Control Transmission Protocol-SCTP, as their transport layer protocol) running on the target system.

Counter measure: The NMap tool delivers specially crafted probes to a target machine. Blocking the ICMP messages is only one of an array of defenses required for full protection against fingerprint attacks [136], [137]. In addition, a fingerprint scrubber is used. A TCP fingerprint scrubber is a tool that prevents a remote user from determining the operating system of another host on a network. It works at both the network and transport layers to convert ambiguous traffic [138], [139]. The fingerprint scrubber is built on the TCP scrubber and removes ambiguities from flows that can reveal implementation-specific details. TCP/IP fingerprinting involves detecting all open TCP and UDP ports to determine which services are running on the host [140]. The default scan is approximately 1900 TCP ports and 180 UDP ports.

5.2. SYN flooding

One of the protocols that exist at the Transport Layer is TCP. TCP is used to establish a connection-oriented session between two devices that want to communication or exchange data.

For every TCP SYN packet received on a device, a TCP ACK packet must be sent back in response. One type of attack that takes advantage of this design flaw in TCP is known as a SYN Flood attack. The attacker sends continuous stream of TCP SYN packets to a target system Uses random source IP addresses are used [141], [142]. This causes the target machine to process each individual packet and respond accordingly. Eventually, with the high influx of TCP SYN packets, the target system will become too overwhelmed and stop responding to any requests [143].

Counter measure: Use a keyed hash (H) Cookie.

Has an algorithm that creates a message authentication code based on both a message and a secret key shared by two endpoints. Also known as a hash message authentication code algorithm. After a server receives a SYN packet, it calculates a keyed hash (H from the information in the packet using a secret key that is only known to the server [144]. This hash (H) is sent to the client as the initial sequence number from the server. H is called SYN cookie. The server will not store the half-open connection in its queue. If the client is an attacker, H will not reach the attacker. If the client is not an attacker, it sends H+1 in the acknowledgement field. The server checks if the number in the acknowledgement field is valid or not by recalculating the cookie [145].

5.3. TCP reassembly and sequencing

During a TCP transmission of datagrams between two devices, the sender tags each packet with a sequence number. This sequence number is used to reassemble the packets back into data. During the transmission of packets, each packet may take a different path to the destination. This may cause the packets to be received in an out-of-order fashion, or in the order, the sender sent them [146]. An attacker can attempt to guess the sequencing numbers of packets and inject malicious packets into the network destined for the target. When the target receives the packets, the receiver would assume they came from the real sender, as they would contain the appropriate sequence numbers and a spoofed IP address [147].

Counter measure: Timing differences or information from lower(Data Link, Network) protocol layers could allow the receiving host to distinguish authentic TCP packets from the sending host and counterfeit TCP packets with the correct sequence number sent by the attacker [148]. If such other information is available to the receiving host, if the attacker can also fake that other information, and if the receiving host gathers and uses the information correctly, then the receiving host may be fairly immune to TCP sequence prediction attacks. Usually, TCP sequence number is the primary means of protection of TCP traffic against these types of attack.

5.4. IP Spoofing

IP address spoofing or IP spoofing refers to the creation of Internet Protocol (IP) packets with a forged source IP address, called spoofing, with the purpose of concealing, the identity of the sender or impersonating another computing system [149], [150]. Figure 7 demonstrates how IP spoofing works. Attackers may generate fraudulent packet headers, continuously randomizing the source address using a sniffing tool. They may also use the IP address of another existing device so that responses to the spoofed packet go there instead.

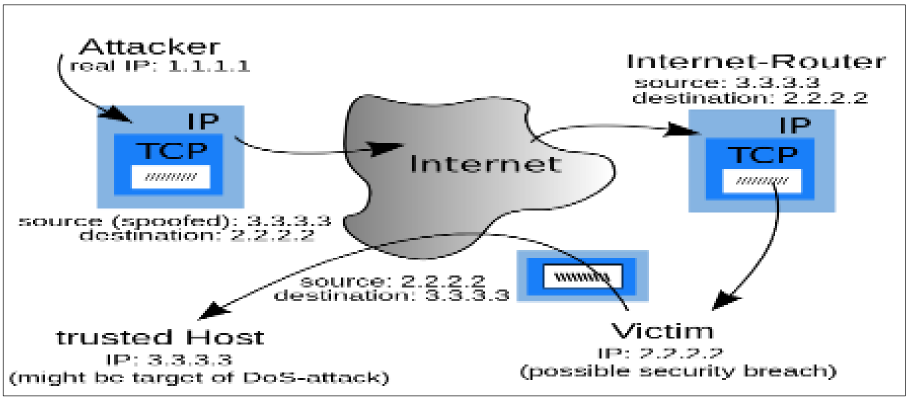

Figure 7 IP Spoofing in DoS attacks

Counter Measures: The first step is to eliminate host-based authentication on your network. Host-based authentication uses the public host key of the client machine to authenticate a user [151]. Second, use Ingress filtering, a technique, which verifies that packets are coming from a legitimate source, is also an invaluable tool to safeguard against attacks perpetuated through IP spoofing. Third, use Egress filtering, in which packets that are being sent out of the internal network are examined via router or firewall and questionable packets are detained, and is often used in conjunction with ingress filtering [152]-[154]. Fourth, use a proxy Server to hide your IP address, verifying traffic, and blocking access by unauthorized outsiders. Finally, use a VPN. Your internet traffic data will be sent to the VPN via a secure connection [155] and routed appropriately to the sites you intend to visit, effectively making your own IP address private and hidden.

5.5. TCP session hijacking

TCP session hijacking is a malicious technique that exploits the way TCP (Transmission Control Protocol) works to take over an established connection between two devices on a network [156], [157]. By hijacking a TCP session, an attacker can impersonate one of the parties, intercept or alter the data, or launch other attacks. Transport Layer Hijacking occurs in TCP sessions and involves the attacker disrupting the communication channel between a client and server in such a way that data is unable to be exchanged [158].

Counter Measure: In order to protect against TCP session hijacking attacks, it is important to secure your network and devices from unauthorized access and monitoring [159]-[161]. Encryption and authentication protocols, employed to protect the data and identity of the endpoints.

5.6. RST and FIN denial of service attack

RST (Reset) and **FIN (Finish)** denial of service (DoS) attacks are types of attacks that exploit vulnerabilities in the TCP protocol at the transport layer [162]. These attacks aim to disrupt network communication by sending forged TCP packets to terminate existing connections or reset connections, thereby preventing legitimate users from accessing services.

RST Attack: In an RST attack, the attacker sends a TCP RST packet to one or both endpoints of a TCP connection, with the goal of terminating the connection abruptly. This can lead to a denial of service for legitimate users, as their connections are unexpectedly closed, causing data loss and disruption of services.

FIN Attack: In a FIN attack, the attacker sends a TCP FIN packet to one or both endpoints of a TCP connection, indicating that the sender has finished sending data. This can be used maliciously to trick the endpoints into closing the connection, causing disruption to legitimate users.

Countermeasures: The following are some of the solutions to RST and FIN denial of service attacks

Firewalls and Intrusion Detection Systems (IDS): Implement firewalls and IDS to detect and block malicious TCP packets, including RST and FIN packets.

TCP Sequence Number Randomization: Randomize TCP sequence numbers to make it harder for attackers to predict and forge TCP packets.

Rate Limiting: Implement rate limiting to prevent an excessive number of TCP packets from a single source, which can help mitigate the impact of DoS attacks.

TCP Stateful Inspection: Use TCP stateful inspection to validate incoming TCP packets and ensure they are part of legitimate connections.

Network Traffic Monitoring: Continuously monitor network traffic for signs of unusual or malicious activity, which can help detect and mitigate DoS attacks in real-time.

5.7. Ping of Death(PoD)

A Ping of Death (PoD) attack is a form of DDoS attack in which an attacker sends the recipient simple ping requests as fragmented IP packets that are oversized or malformed [163], [164]. These packets do not adhere to the IP packet format when reassembled, leading to heap/memory errors and system crashes.

Counter Measure: Configure your firewall, add filters, look at spoofed packets, monitoring traffic patterns, and frequently scan the network [165].

5.8. Low Rate/ Shrew Attacks

These are DDoS attacks that generate periodic, short bursts of high volume traffic and create congestion [166], [167]. This forces legitimate TCP connections to drastically reduce their sending rate. Figure 8 shows a typical shrew attack which exploits the deficiencies in the retransmission time-out (RTO) mechanism of TCP flows. They throttle legitimate TCP flows by periodically sending burst pulses with high peak rate in a low frequency [168]. As such, the TCP flows see congestion on the attacked link every time it recovers from RTO. Indeed, such a shrew attack may reduce the throughput of TCP applications down to almost zero.

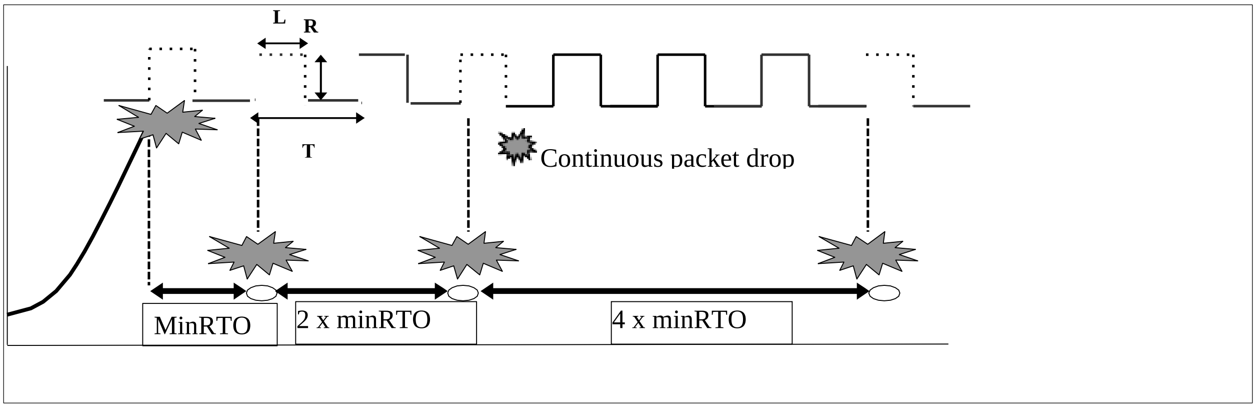

Figure 8 A Shrew attack

Counter Measure: A simple protection mechanism called SAP (Shrew Attack Protection) can be used to defend against a shrew attack. As shown in Figure 9, SAP is a destination-port-based mechanism that only requires a small number of counters. TCP uses packet drops as an indication of congestion and reacts to a packet drop by reducing the rate of the corresponding flow [169]-[171]. The main idea of SAP is to neutralize a Shrew attack by controlling the drop rates of TCP flows at the application- aggregate level via the use of differential packet prioritization.

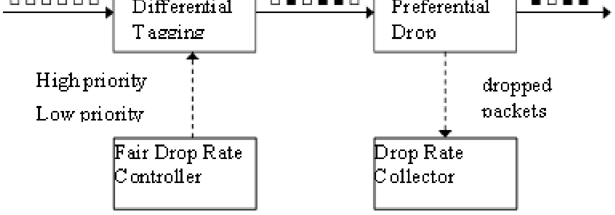

Figure 9 SAP Architecture

Destination port in the TCP/IP header of each packet is used to identify the application aggregate. The drop rates of application-aggregates, based on which SAP identifies potential victims are monitored by Drop Rate Collector. Note that SAP can easily generalize it to other aggregation levels. Alternatively, as often used in modern routers, SAP can employ a hash of flow description fields in the packet. While SAP also can consider using different fair drop rates for different types of packets. After the fair drop rate is determined, SAP starts to protect the victims by tagging their TCP packets as high priority to lower the victims' drop rate (e.g., controlled by Differential Tagging module) if their drop rates grow higher than the fair drop rate. Otherwise, they will be tagged as normal (e.g., low priority). All tagged packets will be passed to the priority Active Queue Management (AQM) module in the router, which implements preferential packet dropping [172], [173]. Note that SAP could be treated as a form of traffic management mechanism that aims to ensure all flows experience similar drop rates when going through the same protocol by using multiple classes/tagging on flow level.

6. Discussion

It has been shown that the transport layer of the TCP/IP stack plays a crucial role in ensuring the efficient, reliable, and secure transmission of data across networks. Performance issues at this layer can arise due to factors such as network congestion, latency, and packet loss. TCP, being a connection-oriented protocol, employs mechanisms like flow control and congestion avoidance to manage these issues [174]. However, these mechanisms can sometimes lead to performance degradation, especially in high-latency or high-loss network environments. Additionally, the overhead [175] introduced by TCP's reliability mechanisms, such as acknowledgment messages and retransmission of lost packets, can impact performance, particularly in scenarios where real-time communication or high throughput is required [176], [177]. To mitigate these performance issues, optimization techniques such as TCP window scaling, selective acknowledgment, and congestion control algorithms like TCP Cubic are employed to adapt TCP's behavior dynamically to network conditions, optimizing throughput and minimizing latency. Table 2 presents a summary of the performance issues in the TCP/IP protocol suite.

Table 2 TCP performance issues

Issues	Description
Throughput	The maximum rate at which data can be transmitted over a network may be limited by the transport layer protocol or network congestion
Latency	The time delay between sending and receiving data packets can impact real-time applications like video conferencing and online gaming
Packet Loss	Occurs when data packets are lost during transmission, often due to network congestion or errors, leading to retransmissions and reduced throughput
Privacy Issues	Data Interception: Attackers can intercept and eavesdrop on data transmitted over the network, compromising the confidentiality of the information
Data Tampering	Attackers can modify or alter data packets in transit, leading to integrity issues and potential security risks
Traffic Analysis	By analyzing the patterns and volume of network traffic, attackers can glean sensitive information about the communication patterns of users

Privacy concerns at the transport layer primarily revolve around the security and confidentiality of data transmitted between communicating parties [178]-[181]. Without proper encryption mechanisms, data sent over TCP/IP networks can be intercepted and accessed by unauthorized parties, compromising user privacy. Transport Layer Security (TLS), which operates at the transport layer, addresses these concerns by providing end-to-end encryption and authentication for data transmitted between clients and servers. By encrypting data in transit, TLS protects sensitive information from eavesdropping and interception, ensuring user privacy and confidentiality [182], [183]. However, implementation flaws or misconfigurations in TLS can sometimes lead to vulnerabilities, undermining its effectiveness in protecting user privacy. Additionally, privacy concerns may arise from the collection and storage of metadata associated with TCP/IP connections, such as IP addresses, port numbers, and timestamps, which can be used to track and profile users' online activities.

Security issues at the transport layer encompass a wide range of threats, including session hijacking, man-in-the-middle attacks, and denial-of-service (DoS) attacks. TCP/IP protocols like TCP and UDP are vulnerable to these attacks due to their connection-oriented and connectionless nature, respectively [184]-[187]. TCP-based attacks, such as SYN flooding and TCP reset attacks, exploit weaknesses in the TCP handshake process to overwhelm target systems with a high volume of malicious traffic, causing service disruptions or denial of service. UDP-based attacks, such as UDP flooding and DNS amplification attacks, leverage the stateless nature of UDP to flood target systems with spoofed packets, consuming network resources and disrupting service availability. To mitigate these security issues, network administrators can implement security measures such as stateful firewalls, intrusion detection systems (IDS) [188], and rate limiting to detect and mitigate malicious traffic targeting TCP/IP protocols.

Furthermore, the transport layer is vulnerable to protocol-specific attacks that exploit weaknesses in TCP/IP implementations or configurations. For example, vulnerabilities in TCP's handling of sequence numbers or window sizes can be exploited to manipulate TCP connections and compromise network security [189], [190]. Similarly, flaws in UDP implementations can lead to amplification attacks or enable unauthorized access to network services. Table 3 illustrates some of the privacy and security concerns in the TCP/IP protocol suite.

Table 3 TCP/IP Security and privacy concerns

Issues	**Discussion**
Denial of Service (DoS) Attacks	Attackers can flood a network or server with excessive traffic, causing it to become unavailable to legitimate users
SYN Flood Attacks	Attackers send a large number of TCP SYN requests to a server, overwhelming its resources and making it unable to respond to legitimate requests
Session Hijacking	Attackers take over an ongoing session between two parties, gaining unauthorized access to the session's resources
RST and FIN denial of service attack	RST (Reset) and FIN (Finish) denial of service (DoS) attacks are types of attacks that exploit vulnerabilities in the TCP protocol at the transport layer. These attacks aim to disrupt network communication by sending forged TCP packets to terminate existing connections or reset connections, thereby preventing legitimate users from accessing services RST Attack: In an RST attack, the attacker sends a TCP RST packet to one or both endpoints of a TCP connection, with the goal of terminating the connection abruptly. This can lead to a denial of service for legitimate users, as their connections are unexpectedly closed, causing data loss and disruption of services FIN Attack: In a FIN attack, the attacker sends a TCP FIN packet to one or both endpoints of a TCP connection, indicating that the sender has finished sending data. This can be used maliciously to trick the endpoints into closing the connection, causing disruption to legitimate users.
Finger Printing	Used to discover open ports and services that are running open on the target system
Low Rates/Shrew Attacks	DDoS attack that generate periodic, short bursts of high volume traffic and create congestion. This forces legitimate TCP connections to reduce their sending rate. Shrew attacks exploit the deficiencies in the retransmission time-out (RTO) mechanism of TCP flows
Ping of death attack	DDoS attack in which an attacker sends the recipient simple ping requests as fragmented IP packets that are oversized or malformed. These packets do not adhere to the IP packet format when reassembled, leading to heap/memory errors and system crashes.
IP Spoofing	Creation of Internet Protocol (IP) packets with a forged source IP address, called spoofing, with the purpose of concealing, the identity of the sender or impersonating another computing system Attackers may generate fraudulent packet headers, continuously randomizing the source address using a sniffing tool

To address these vulnerabilities, software vendors release patches and updates to fix known security issues, and network administrators apply these patches promptly to protect against potential exploits. Additionally, security awareness training and best practices in network configuration and management are essential for preventing and mitigating security incidents at the transport layer of the TCP/IP stack. Table 4 details some of the countermeasures that can be deployed to address these concerns.

Table 4 TCP/IP Security and privacy concerns countermeasures

Countermeasures	**Discussion**
Encryption	Use of protocols like TLS (Transport Layer Security) to encrypt data in transit, ensuring confidentiality
Rate Limiting	Implement rate limiting to prevent an excessive number of TCP packets from a single source, which can help mitigate the impact of DoS attacks.
TCP Stateful Inspection:	Use TCP stateful inspection to validate incoming TCP packets and ensure they are part of legitimate connections
Firewalls	Implement firewalls to filter and monitor incoming and outgoing network traffic, protecting against unauthorized access and DoS attacks
Intrusion Detection Systems (IDS)	Deploy IDS to detect and respond to suspicious network activity, mitigating potential security threats
Quality of Service (QoS)	Implement QoS mechanisms to prioritize and manage network traffic, ensuring optimal performance for critical applications
Network Traffic Monitoring	Continuously monitor network traffic for signs of unusual or malicious activity, which can help detect and mitigate DoS attacks in real-time
Keyed hash (H) Cookie	Prevents Syn Flood attacks
TCP Sequence Number Randomization	Randomize TCP sequence numbers to make it harder for attackers to predict and forge TCP packets
Network Traffic Monitoring	Continuously monitor network traffic for signs of unusual or malicious activity, which can help detect and mitigate DoS attacks in real-time.

6.1. Research Gaps

Research in the area of performance, privacy, and security issues at the transport layer of the TCP/IP stack has made significant progress, but there are still gaps that researchers are actively working to address. Some of these gaps include:

Emerging Protocols: With the advent of new transport layer protocols such as QUIC (Quick UDP Internet Connections) [191], there is a need for research to evaluate their performance, privacy, and security implications compared to traditional protocols like TCP and UDP.

Machine Learning Applications: There is a growing interest in leveraging machine learning techniques to enhance the performance, privacy, and security of transport layer protocols [192], [193]. Research in this area aims to develop intelligent algorithms that can adapt to changing network conditions and mitigate security threats.

Privacy-preserving Protocols: As privacy concerns become increasingly important, there is a need for research to develop transport layer protocols that can ensure the confidentiality and integrity of data without compromising performance. According to [194], privacy-preserving protocols are designed to enable secure communication and data exchange while minimizing the exposure of sensitive information. These protocols typically employ cryptographic techniques to protect the confidentiality, integrity, and authenticity of data transmitted over networks [195]-[199]. One example is Secure Multi-Party Computation (SMPC), which allows multiple parties to jointly compute a function over their inputs while keeping those inputs private. Another example is Homomorphic Encryption, which enables computations to be performed on encrypted data without decrypting it, preserving privacy even during data processing. Additionally, protocols like Zero-Knowledge Proofs and Differential Privacy provide mechanisms for verifying information or performing data analysis without revealing sensitive details about individuals. These privacy-preserving protocols play a crucial role in ensuring user privacy and data protection in various applications, including healthcare, finance, and telecommunications.

Quantum-safe Cryptography: With the emergence of quantum computing, there is a need to develop transport layer protocols that are resistant to quantum attacks [200]. Research in this area focuses on developing quantum-safe cryptographic algorithms and protocols.

Cross-layer Optimization: There is a need for research to explore cross-layer optimization techniques that can improve the overall performance, privacy, and security of the TCP/IP stack by considering interactions between different layers of the protocol stack [201].

Energy Efficiency: With the proliferation of mobile and IoT devices, there is a growing need for transport layer protocols that are energy-efficient [202]. Research in this area focuses on developing protocols that can reduce energy consumption without compromising performance or security.

Standardization and Interoperability: As new transport layer protocols and technologies emerge, there is a need for research to address standardization and interoperability issues to ensure seamless communication between different networks and devices [204].

6.2. Future Research Scope

Future research in the area of performance, privacy, and security issues at the transport layer of the TCP/IP stack is expected to focus on several key areas. Some of the potential future research scope includes:

5G and Beyond: With the deployment of 5G networks [204] and the ongoing development of future generations of wireless networks, there is a need for research to address the unique performance, privacy, and security challenges posed by these networks at the transport layer.

Internet of Things (IoT): As the number of IoT devices continues to grow, there is a need for research to develop transport layer protocols [205] that can efficiently handle the communication requirements of IoT devices while ensuring privacy and security.

Edge Computing: With the increasing adoption of edge computing [206], there is a need for research to develop transport layer protocols that can efficiently support communication between edge devices and the cloud while ensuring low latency and high reliability.

AI-driven Networking: With the growing use of artificial intelligence (AI) in networking [207], there is a need for research to explore how AI can be used to optimize the performance, privacy, and security of transport layer protocols.

Blockchain and Distributed Ledger Technologies: With the increasing interest in blockchain and distributed ledger technologies [208], there is a need for research to explore how these technologies can be used to enhance the privacy and security of transport layer protocols.

Post-Quantum Cryptography: With the advent of quantum computing [209], there is a need for research to develop post-quantum cryptographic algorithms and protocols that can ensure the security of transport layer communications in a post-quantum computing era.

Software-Defined Networking (SDN) and Network Function Virtualization (NFV): With the increasing adoption of SDN and NFV [210], there is a need for research to explore how these technologies can be used to optimize the performance, privacy, and security of transport layer protocols.

Cross-layer Optimization: There is a need for research to continue exploring cross-layer optimization techniques that can improve the overall performance, privacy, and security of the TCP/IP stack by considering interactions between different layers of the protocol stack.

Privacy-Preserving Protocols: With the increasing concern for privacy, there is a need for research to develop transport layer protocols that can ensure the confidentiality and integrity of data without compromising performance.

Energy-Efficient Protocols: With the proliferation of mobile and IoT devices, there is a need for research to develop transport layer protocols that are energy-efficient while maintaining performance and security.

Addressing these future research scopes will be crucial in ensuring the development of efficient, secure, and privacy-preserving transport layer protocols to support the evolving needs of modern networking environments.

7. Conclusion

The transport layer of the TCP/IP stack plays a critical role in ensuring reliable data transmission between applications. However, it faces various challenges related to performance, privacy, and security. By implementing the right strategies and best practices, these issues can be mitigated, ensuring a secure and efficient network communication environment. TCP/IP at the transport layer can face performance issues due to factors like latency and congestion. Latency can be reduced by using UDP for real-time applications, while congestion can be managed through TCP's congestion control algorithms like TCP Vegas or TCP Reno. Security concerns in TCP/IP at the transport layer include the potential for unauthorized access and data breaches. Implementing encryption protocols like TLS can protect data in transit, while VPNs can create secure tunnels for data to pass through, preventing unauthorized access. Privacy issues in TCP/IP at the transport layer relate to the potential for eavesdropping and data interception. Encryption protocols like TLS can encrypt data, ensuring that sensitive information remains private and secure during transmission.

Compliance with ethical standards

Acknowledgements

I would like to thank all people who offered some support during the writing of this paper.

Disclosure of conflict of interest

The author declares that he holds no conflict of interest.

References

[1] Kumar PR, Wan AT, Suhaili WS. Exploring data security and privacy issues in internet of things based on five-layer architecture. International journal of communication networks and information security. 2020 Apr 1;12(1):108-21.

[2] Al-Jarrah M, Abdel-karim RT. A thin security layer protocol over IP protocol on TCP/IP suite for security enhancement. In2006 Innovations in Information Technology 2006 Nov 19 (pp. 1-5). IEEE.

[3] 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..

[4] Sarkar D, Narayan H. Transport layer protocols for cognitive networks. In2010 INFOCOM IEEE Conference on Computer Communications Workshops 2010 Mar 15 (pp. 1-6). IEEE.

[5] Shieh SP, Ho FS, Huang YL, Luo JN. Network address translators: effects on security protocols and applications in the TCP/IP stack. IEEE Internet computing. 2000 Nov;4(6):42-9.

[6] 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).

[7] Papastergiou G, Fairhurst G, Ros D, Brunstrom A, Grinnemo KJ, Hurtig P, Khademi N, Tüxen M, Welzl M, Damjanovic D, Mangiante S. De-ossifying the internet transport layer: A survey and future perspectives. IEEE Communications Surveys & Tutorials. 2016 Nov 8;19(1):619-39.

[8] Apostolopoulos G, Peris V, Saha D. Transport Layer Security: How much does it really cost?. InIEEE INFOCOM'99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320) 1999 Mar 21 (Vol. 2, pp. 717-725). IEEE.

[9] Verma LP, Sharma VK, Kumar M, Mahanti A. An adaptive multi-path data transfer approach for MP-TCP. Wireless Networks. 2022 Jul;28(5):2185-212.

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

[11] Parween S, Hussain SZ. TCP Performance Enhancement in IoT and MANET: A Systematic Literature Review. International Journal of Computer Networks and Applications. 2023 Aug 31:543-68.

[12] 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.

[13] Lastinec J, Hudec L. Comparative analysis of TCP/IP security protocols for use in vehicle communication. In2016 17th International Carpathian Control Conference (ICCC) 2016 May 29 (pp. 429-433). IEEE.

[14] Hunt C. TCP/IP network administration. " O'Reilly Media, Inc."; 2002.

[15] Coonjah I, Catherine PC, Soyjaudah KM. Experimental performance comparison between TCP vs UDP tunnel using OpenVPN. In2015 International Conference on Computing, Communication and Security (ICCCS) 2015 Dec 4 (pp. 1-5). IEEE.

[16] Nath PB, Uddin MM. Tcp-ip model in data communication and networking. American Journal of Engineering Research. 2015;4(10):102-7.

[17] Apostol GC, Mocanu AE, Mocanu BC, Radulescu DM, Pop F. CPSOCKS: Cross-Platform Privacy Overlay Adapter Based on SOCKSv5 Protocol. InInternational Conference on Green, Pervasive, and Cloud Computing 2022 Dec 2 (pp. 149-161). Cham: Springer International Publishing.

[18] 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.

[19] 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.

[20] Bardhi E, Conti M, Lazzeretti R, Losiouk E. Security and privacy of IP-ICN coexistence: A comprehensive survey. IEEE Communications Surveys & Tutorials. 2023 Jul 13.

[21] Katulić F, Sumina D, Groš S, Erceg I. Protecting Modbus/TCP-Based Industrial Automation and Control Systems Using Message Authentication Codes. IEEE access. 2023 May 11.

[22] Tymchenko O, Havrysh B. Steganography in TCP/IP Networks. InInternational Conference of Artificial Intelligence, Medical Engineering, Education 2022 Aug 19 (pp. 47-56). Cham: Springer Nature Switzerland.

[23] Nachbar JM, Kinney BM, Sacks JM, Gurtner GC, TerKonda SP, Reddy SK, Jeffers LL. Cybersecurity and Technical Patient Privacy Protection. Plastic and Reconstructive Surgery. 2023 May 22:10-97.

[24] Laštovička M, Husák M, Velan P, Jirsík T, Čeleda P. Passive operating system fingerprinting revisited: Evaluation and current challenges. Computer Networks. 2023 Jun 1;229:109782.

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

[26] Pawar AB, Jawale MA, William P, Sonawane BS. Efficacy of TCP/IP Over ATM Architecture Using Network Slicing in 5G Environment. InSmart Data Intelligence: Proceedings of ICSMDI 2022 2022 Aug 18 (pp. 79-93). Singapore: Springer Nature Singapore.

[27] Al-Shareeda MA, Manickam S, Laghari SA, Jaisan A. Replay-attack detection and prevention mechanism in industry 4.0 landscape for secure SECS/GEM communications. Sustainability. 2022 Nov 29;14(23):15900.

[28] Jain A, Bhullar S. Network performance evaluation of smart distribution systems using smart meters with TCP/IP communication protocol. Energy Reports. 2022 Nov 1;8:19-34.

[29] Chen CL, Yang J, Tsaur WJ, Weng W, Wu CM, Wei X. Enterprise data sharing with privacy-preserved based on hyperledger fabric blockchain in IIOT's application. Sensors. 2022 Feb 2;22(3):1146.

[30] Mahamune AA, Chandane MM. TCP/IP layerwise taxonomy of attacks and defence mechanisms in mobile Ad Hoc networks. Journal of The Institution of Engineers (India): Series B. 2022 Feb;103(1):273-91.

[31] 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.

[32] Qiao L, Dong E, Yin H, Li H, Yang J. Intelligent Network Device Identification based on Active TCP/IP Stack Probing. IEEE Network. 2024 Mar 7.

[33] Shirichian M, Sabbaghi-Nadooshan R, Houshmand M, Houshmand M. A QTCP/IP reference model for partially trusted-node-based quantum-key-distribution-secured optical networks. Quantum Information Processing. 2024 Mar 1;23(3):87.

[34] 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.

[35] Kizza JM. Internet of things (iot): growth, challenges, and security. InGuide to Computer Network Security 2024 Jan 20 (pp. 557-573). Cham: Springer International Publishing.

[36] Ghazo AT, Kumar R. ANDVI: Automated Network Device and Vulnerability Identification in SCADA/ICS by Passive Monitoring. IEEE Transactions on Systems, Man, and Cybernetics: Systems. 2024 Jan 15.

[37] 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.

[38] Yoo S, Chen X, Rexford J. SmartCookie: Blocking Large-Scale SYN Floods with a Split-Proxy Defense on Programmable Data Planes. InUSENIX Security 2024.

[39] 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.

[40] Ling X, Yu J, Zhao Z, Zhou Z, Xu H, Chen B, Zhang F. DDoSMiner: An Automated Framework for DDoS Attack Characterization and Vulnerability Mining. InInternational Conference on Applied Cryptography and Network Security 2024 Feb 29 (pp. 283-309). Cham: Springer Nature Switzerland.

[41] Uddin R, Kumar SA, Chamola V. Denial of service attacks in edge computing layers: Taxonomy, vulnerabilities, threats and solutions. Ad Hoc Networks. 2024 Jan 1;152:103322.

[42] ur Rehman S, Khaliq M, Imtiaz SI, Rasool A, Shafiq M, Javed AR, Jalil Z, Bashir AK. DIDDOS: An approach for detection and identification of Distributed Denial of Service (DDoS) cyberattacks using Gated Recurrent Units (GRU). Future Generation Computer Systems. 2021 May 1;118:453-66.

[43] Alsamhi SH, Shvetsov AV, Kumar S, Shvetsova SV, Alhartomi MA, Hawbani A, Rajput NS, Srivastava S, Saif A, Nyangaresi VO. UAV computing-assisted search and rescue mission framework for disaster and harsh environment mitigation. Drones. 2022 Jun 22;6(7):154.

[44] Jaber AN, Anwar S, Khidzir NZ, Anbar M. A detailed analysis on intrusion identification mechanism in cloud computing and datasets. InAdvances in Cyber Security: Second International Conference, ACeS 2020, Penang, Malaysia, December 8-9, 2020, Revised Selected Papers 2 2021 (pp. 550-573). Springer Singapore.

[45] Kozierok CM. The TCP/IP guide: a comprehensive, illustrated Internet protocols reference. No Starch Press; 2005 Oct 1.

[46] Harrison JV, Berghel H. A protocol layer survey of network security. Advances in Computers. 2005 Jan 1;64:109-58.

[47] Wenhua Z, Kamrul Hasan M, Ismail AF, Yanke Z, Razzaque MA, Islam S, Anil kumar B. Data security in smart devices: Advancement, constraints and future recommendations. IET Networks. 2023 Nov;12(6):269-81.

[48] 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.

[49] 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.

[50] Gupta BB, Quamara M. An overview of Internet of Things (IoT): Architectural aspects, challenges, and protocols. Concurrency and Computation: Practice and Experience. 2020 Nov 10;32(21):e4946.

[51] Bhattacharjya A, Zhong X, Wang J, Li X. Security challenges and concerns of Internet of Things (IoT). Cyber-Physical Systems: architecture, security and application. 2019:153-85.

[52] Sadique KM, Rahmani R, Johannesson P. Towards security on internet of things: applications and challenges in technology. Procedia Computer Science. 2018 Jan 1;141:199-206.

[53] Malhotra P, Singh Y, Anand P, Bangotra DK, Singh PK, Hong WC. Internet of things: Evolution, concerns and security challenges. Sensors. 2021 Mar 5;21(5):1809.

[54] Florea I, Ruse LC, Rughinis R. Challenges in security in Internet of Things. In2017 16th RoEduNet Conference: Networking in Education and Research (RoEduNet) 2017 Sep 21 (pp. 1-5). IEEE.

[55] 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).

[56] AlSabah M, Goldberg I. PCTCP: per-circuit TCP-over-IPsec transport for anonymous communication overlay networks. InProceedings of the 2013 ACM SIGSAC conference on Computer & communications security 2013 Nov 4 (pp. 349-360).

[57] Hsu FH, Hwang YL, Tsai CY, Cai WT, Lee CH, Chang K. TRAP: A three-way handshake server for TCP connection establishment. Applied Sciences. 2016 Nov 16;6(11):358.

[58] Lopez JA, Sun Y, Blair PB, Mukhtar MS. TCP three-way handshake: linking developmental processes with plant immunity. Trends in plant science. 2015 Apr 1;20(4):238-45.

[59] 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.

[60] Rahouma KH, Abdul-Karim MS, Nasr KS. TCP/IP Network Layers and Their Protocols (A Survey). InInternet of Things—Applications and Future: Proceedings of ITAF 2019 2020 (pp. 287-323). Springer Singapore.

[61] 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.

[62] Kutscher D. It's the network: Towards better security and transport performance in 5G. In2016 IEEE conference on computer communications workshops (INFOCOM WKSHPS) 2016 Apr 10 (pp. 656-661). IEEE.

[63] Fan X, Gou G, Kang C, Shi J, Xiong G. Identify OS from encrypted traffic with TCP/IP stack fingerprinting. In2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC) 2019 Oct 29 (pp. 1-7). IEEE.

[64] Radhakrishnan S, Cheng Y, Chu J, Jain A, Raghavan B. TCP fast open. InProceedings of the Seventh COnference on emerging Networking EXperiments and Technologies 2011 Dec 6 (pp. 1-12).

[65] Kumar MA, Karthikeyan S. Security model for TCP/IP protocol suite. Journal of Advances in Information Technology. 2011 May;2(2):87-91.

[66] Siriwardena P, Siriwardena P. How Transport Layer Security Works?. Advanced API Security: OAuth 2.0 and Beyond. 2020:355-76.

[67] 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.

[68] Coonjah I, Catherine PC, Soyjaudah KM. Experimental performance comparison between TCP vs UDP tunnel using OpenVPN. In2015 International Conference on Computing, Communication and Security (ICCCS) 2015 Dec 4 (pp. 1-5). IEEE.

[69] Sing J, Soh B. A critical analysis of multilayer IP security protocol. InThird International Conference on Information Technology and Applications (ICITA'05) 2005 Jul 4 (Vol. 2, pp. 683-688). IEEE.

[70] Verma DA, Singh V, Shree R, Minhas D, Yuvaraj S. An Analysis of Encapsulating Security Payload in Wireless Network Security. In2023 International Conference on Emerging Research in Computational Science (ICERCS) 2023 Dec 7 (pp. 1-6). IEEE.

[71] Rochet F, Assogba E, Piraux M, Edeline K, Donnet B, Bonaventure O. TCPLS: Modern transport services with TCP and TLS. InProceedings of the 17th International Conference on emerging Networking EXperiments and Technologies 2021 Dec 2 (pp. 45-59).

[72] Zaman S, Karray F. TCP/IP model and intrusion detection systems. In2009 International Conference on Advanced Information Networking and Applications Workshops 2009 May 26 (pp. 90-96). IEEE.

[73] 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.

[74] Shirsath VA, Chandane MM, Lal C, Conti M. SPARQ: SYN Protection using Acyclic Redundancy check and Quartile range on P4 switches. Computer Communications. 2024 Feb 15;216:283-94.

[75] Kumar P, Dezfouli B. quicSDN: Transitioning from TCP to QUIC for southbound communication in software-defined networks. Journal of Network and Computer Applications. 2024 Feb 1;222:103780.

[76] Mateen AH, Zhu Q, Afsar S, Sahil SA. Effect of encryption delay on TCP and UDP transport layer protocols in software defined networks (SDN). Inthe Proceedings of the International MultiConference of Engineers and Computer Scientists, Hong kong, Hong kong 2019 Mar (pp. 13-15).

[77] Mahboob A, Ikram N. Transport Layer Security (TLS)–A Network Security Protocol for E-commerce. Pakistan Navy Engineering College (PNEC) Research Journal. 2004.

[78] Loshin P. TCP/IP clearly explained. Elsevier; 2003 Jan 4.

[79] Cruickshank H, Mort R, Berioli M. Broadband Satellite Multimedia (BSM) security architecture and interworking with performance enhancing proxies. InPersonal Satellite Services: International Conference, PSATS 2009, Rome, Italy, March 18-19, 2009, Revised Selected Papers 1 2009 (pp. 132-142). Springer Berlin Heidelberg.

[80] Bhutta MN, Cruickshank H, Ashworth J, Moseley M. Redesigning of IPSec for interworking with satellite Performance Enhancing Proxies. In2011 6th International ICST Conference on Communications and Networking in China (CHINACOM) 2011 Aug 17 (pp. 1104-1109). IEEE.

[81] Smith FD, Campos FH, Jeffay K, Ott D. What TCP/IP protocol headers can tell us about the web. InProceedings of the 2001 ACM SIGMETRICS international conference on Measurement and modeling of computer systems 2001 Jun 1 (pp. 245-256).

[82] Dunkels A, Alonso J, Voigt T, Ritter H, Schiller J. Connecting wireless sensornets with TCP/IP networks. InWired/Wireless Internet Communications: Second International Conference, WWIC 2004, Frankfurt (Oder), Germany, February 4-6, 2004. Proceedings 2 2004 (pp. 143-152). Springer Berlin Heidelberg.

[83] Abed AE. Hiding Text in Sequence Number Field of TCP/IP (Doctoral dissertation, Ministry of Higher Education), 2017.

[84] 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.

[85] Chappell L. Inside the tcp handshake. NetWare Connection. 2000 Mar.

[86] Floyd S. Inappropriate TCP resets considered harmful. 2002 Aug.

[87] Bhutta MN, Cruickshank H, Ashworth J, Moseley M. Redesigning of IPSec for interworking with satellite Performance Enhancing Proxies. In2011 6th International ICST Conference on Communications and Networking in China (CHINACOM) 2011 Aug 17 (pp. 1104-1109). IEEE.

[88] Garcia-Macias JA. Transport in the IP-based Internet of Things: status report. Procedia Computer Science. 2023 Jan 1;224:18-25.

[89] Balan RK, Lee BP, Kumar KR, Jacob L, Seah WK, Ananda AL. TCP HACK: TCP header checksum option to improve performance over lossy links. InProceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No. 01CH37213) 2001 Apr 22 (Vol. 1, pp. 309-318). IEEE.

[90] Cil AE, Yildiz K, Buldu A. Detection of DDoS attacks with feed forward based deep neural network model. Expert Systems with Applications. 2021 May 1;169:114520.

[91] 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.

[92] Furutani N, Ban T, Nakazato J, Shimamura J, Kitazono J, Ozawa S. Detection of DDoS backscatter based on traffic features of darknet TCP packets. In2014 Ninth Asia Joint conference on information security 2014 Sep 3 (pp. 39-43). IEEE.

[93] Khan S, Gani A, Wahab AW, Shiraz M, Ahmad I. Network forensics: Review, taxonomy, and open challenges. Journal of Network and Computer Applications. 2016 May 1;66:214-35.

[94] Ferst MK, de Figueiredo HF, Denardin G, Lopes J. Implementation of secure communication with modbus and transport layer security protocols. In2018 13th IEEE International Conference on Industry Applications (INDUSCON) 2018 Nov 12 (pp. 155-162). IEEE.

[95] Varadhan S. Securing Traffic Tunnelled over TCP or UDP. Texas, US: Oracle Corporation. 2016.

[96] Osanaiye OA, Dlodlo M. TCP/IP header classification for detecting spoofed DDoS attack in Cloud environment. InIEEE EUROCON 2015-International Conference on Computer as a Tool (EUROCON) 2015 Sep 8 (pp. 1-6). IEEE.

[97] 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).

[98] Garcia J, Brunstrom A. Checksum-based loss differentiation. In4th International Workshop on Mobile and Wireless Communications Network 2002 Sep 9 (pp. 244-248). IEEE.

[99] Jacobson V. Compressing TCP/IP headers for low-speed serial links. 1990 Feb.

[100] M. Pohl and J. Kubela, "Performance evaluation of application layer protocols for the internet-of-things," in 2018 Sixth International Conference on Enterprise Systems (ES), IEEE, 2018, pp. 180-187.

[101] Pohl M, Kubela J, Bosse S, Turowski K. Performance evaluation of application layer protocols for the internet-of-things. In2018 Sixth International Conference on Enterprise Systems (ES) 2018 Oct 1 (pp. 180-187). IEEE.

[102] Floyd S. A report on recent developments in TCP congestion control. IEEE Communications Magazine. 2001 Apr;39(4):84-90.

[103] Jamal H, Sultan K. Performance analysis of tcp congestion control algorithms. International journal of computers and communications. 2008;2(1):30-8.

[104] 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.

[105] Wang J, Wen J, Zhang J, Han Y. TCP-FIT: An improved TCP congestion control algorithm and its performance. In2011 Proceedings IEEE INFOCOM 2011 Apr 10 (pp. 2894-2902). IEEE.

[106] Jacobson V. Congestion avoidance and control. ACM SIGCOMM computer communication review. 1988 Aug 1;18(4):314-29.

[107] Mathis M, Semke J, Mahdavi J, Ott T. The macroscopic behavior of the TCP congestion avoidance algorithm. ACM SIGCOMM Computer Communication Review. 1997 Jul 1;27(3):67-82.

[108] J. a. N. A. a. R. I. Martin, "Delay-based congestion avoidance for TCP," IEEE/ACM Transactions on networking, vol. 11, no. 3, pp. 356--369, 2003.

[109] L. S. a. O. S. W. a. P. L. L. Brakmo, "TCP Vegas: New techniques for congestion detection and avoidance," in Proceedings of the conference on Communications architectures, protocols and applications, 1994, pp. 24--35.

[110] Martin J, Nilsson A, Rhee I. Delay-based congestion avoidance for TCP. IEEE/ACM Transactions on networking. 2003 Jun;11(3):356-69.

[111] 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.

[112] Gogic A, Suljanovic N, Hasanovic A, Mujcic A, Zajc M. Stochastic markov model for TCP throughput. InProc. EUROSIM 2010, 7th EUROSIM Congress on Modelling and Simulation 2010 (p. 23).

[113] Mo J, La RJ, Anantharam V, Walrand J. Analysis and comparison of TCP Reno and Vegas. InIEEE INFOCOM'99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320) 1999 Mar 21 (Vol. 3, pp. 1556-1563). IEEE.

[114] Patel S, Shukla Y, Kumar N, Sharma T, Singh K. A comparative performance analysis of tcp congestion control algorithms: Newreno, westwood, veno, bic, and cubic. In2020 6th International Conference on Signal Processing and Communication (ICSC) 2020 Mar 5 (pp. 23-28). IEEE.

[115] Ha S, Rhee I. Taming the elephants: New TCP slow start. Computer Networks. 2011 Jun 23;55(9):2092-110.

[116] 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.

[117] Sundaresan S, Allman M, Dhamdhere A, Claffy K. TCP congestion signatures. InProceedings of the 2017 Internet Measurement Conference 2017 Nov 1 (pp. 64-77).

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

[119] 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.

[120] 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.

[121] Hu J, Zeng C, Wang Z, Zhang J, Guo K, Xu H, Huang J, Chen K. Load Balancing With Multi-Level Signals for Lossless Datacenter Networks. IEEE/ACM Transactions on Networking. 2024 Feb 22.

[122] Sood R, Kang SS. Hybrid Congestion Control Mechanism in Software Defined Networks. International Journal of Intelligent Systems and Applications in Engineering. 2024;12(6s):686-76.

[123] Xiao S, Deng L, Li S, Wang X. Integrated tcp/ip protocol software testing for vulnerability detection. In2003 International Conference on Computer Networks and Mobile Computing, 2003. ICCNMC 2003. 2003 Oct 20 (pp. 311-319). IEEE.

[124] Paliwal G, Mudgal AP, Taterh S. A study on various attacks of tcp/ip and security challenges in manet layer architecture. InProceedings of Fourth International Conference on Soft Computing for Problem Solving: SocProS 2014, Volume 2 2015 (pp. 195-207). Springer India.

[125] Acharya S, Tiwari N. Survey of DDoS attacks based on TCP/IP protocol vulnerabilities. IOSR Journal of Computer Engineering. 2016 May;18(3):68-76.

[126] Alqahtani AH, Iftikhar M. TCP/IP attacks, defenses and security tools. International Journal of Science and Modern Engineering (IJISME). 2013 Sep;1(10):42-7.

[127] Beverly R. A robust classifier for passive TCP/IP fingerprinting. InInternational workshop on passive and active network measurement 2004 Apr 19 (pp. 158-167). Berlin, Heidelberg: Springer Berlin Heidelberg.

[128] 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.

[129] Naik N, Jenkins P, Savage N, Yang L. A computational intelligence enabled honeypot for chasing ghosts in the wires. Complex & Intelligent Systems. 2021 Feb;7(1):477-94.

[130] Essaadi D, Laassiri J, Hanaoui S. Using NMAP for Data Collection in Cloud Platform. InAdvanced Intelligent Systems for Sustainable Development (AI2SD'2019) Volume 4-Advanced Intelligent Systems for Applied Computing Sciences 2020 (pp. 507-517). Springer International Publishing.

[131] Kumar R, Tlhagadikgora K. Internal network penetration testing using free/open source tools: Network and system administration approach. InAdvanced Informatics for Computing Research: Second International Conference, ICAICR 2018, Shimla, India, July 14–15, 2018, Revised Selected Papers, Part II 2 2019 (pp. 257-269). Springer Singapore.

[132] Elejla OE, Belaton B, Anbar M, Alijla BO. IPv6 OS fingerprinting methods. InAdvances in Visual Informatics: 5th International Visual Informatics Conference, IVIC 2017, Bangi, Malaysia, November 28–30, 2017, Proceedings 5 2017 (pp. 661-668). Springer International Publishing.

[133] Aksoy A, Valle L, Kar G. Automated Network Incident Identification through Genetic Algorithm-Driven Feature Selection. Electronics. 2024 Jan 9;13(2):293.

[134] Abood EW, Abdullah AM, Al Sibahe MA, Abduljabbar ZA, Nyangaresi VO, Kalafy SA, Ghrabta MJ. Audio steganography with enhanced LSB method for securing encrypted text with bit cycling. Bulletin of Electrical Engineering and Informatics. 2022 Feb 1;11(1):185-94.

[135] Yang K, Li Q, Wang H, Sun L, Liu J. Fingerprinting Industrial IoT devices based on multi-branch neural network. Expert Systems with Applications. 2024 Mar 15;238:122371.

[136] Sosnowski M, Zirngibl J, Sattler P, Carle G, Grohnfeldt C, Russo M, Sgandurra D. EFACTLS: Effective Active TLS Fingerprinting for Large-scale Server Deployment Characterization. IEEE Transactions on Network and Service Management. 2024 Feb 9.

[137] BS S, Nagapadma R. P-DNN: Parallel DNN based IDS framework for the detection of IoT vulnerabilities. Security and Privacy. 2024 Jan;7(1):e330.

[138] Yang W, Fang Y, Zhou X, Shen Y, Zhang W, Yao Y. Networked Industrial Control Device Asset Identification Method Based on Improved Decision Tree. Journal of Network and Systems Management. 2024 Apr;32(2):32.

[139] Zhao Z, Li Z, Zhou Z, Yu J, Song Z, Xie X, Zhang F, Zhang R. DDoS family: A novel perspective for massive types of DDoS attacks. Computers & Security. 2024 Mar 1;138:103663.

[140] Bensaid R, Labraoui N, Abba Ari AA, Maglaras L, Saidi H, Abdu Lwahhab AM, Benfriha S. Toward a real-time tcp syn flood ddos mitigation using adaptive neuro-fuzzy classifier and sdn assistance in fog computing. Security and Communication Networks. 2023 Nov 27;2024.

[141] 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.

[142] Eddy W. TCP SYN flooding attacks and common mitigations. 2007 Aug.

[143] Alizai ZA, Tahir H, Murtaza MH, Tahir S, Mcdonald-Maier K. Key-based cookie-less session management framework for application layer security. IEEE Access. 2019 Sep 11;7:128544-54.

[144] Eddy WM. Defenses against TCP SYN flooding attacks. The Internet Protocol Journal. 2006 Dec 4;9(4):2-16.

[145] Dharmapurikar S, Paxson V. Robust TCP Stream Reassembly in the Presence of Adversaries. InUSENIX Security Symposium 2005 Aug (pp. 65-80).

[146] Zhang M, Karp B, Floyd S, Peterson L. RR-TCP: a reordering-robust TCP with DSACK. In11th IEEE International Conference on Network Protocols, 2003. Proceedings. 2003 Nov 4 (pp. 95-106). IEEE.

[147] Hanaoka M, Shimamura M, Kono K. TCP reassembler for layer7-aware network intrusion detection/prevention systems. IEICE transactions on information and systems. 2007 Dec 1;90(12):2019-32.

[148] Nyangaresi VO, Abduljabbar ZA, Refish SH, Al Sibahee MA, Abood EW, Lu S. Anonymous Key Agreement and Mutual Authentication Protocol for Smart Grids. InCognitive Radio Oriented Wireless Networks and Wireless Internet: 16th EAI International Conference, CROWNCOM 2021, Virtual Event, December 11, 2021, and 14th EAI International Conference, WiCON 2021, Virtual Event, November 9, 2021, Proceedings 2022 Mar 31 (pp. 325-340). Cham: Springer International Publishing.

[149] Hanaoka M, Kono K, Shimamura M, Yamaguchi S. An efficient TCP reassembler mechanism for layer7-aware network intrusion detection/prevention systems. In2007 12th IEEE Symposium on Computers and Communications 2007 Jul 1 (pp. 79-86). IEEE.

[150] Zhang C, Hu G, Chen G, Sangaiah AK, Zhang PA, Yan X, Jiang W. Towards a SDN-based integrated architecture for mitigating IP spoofing attack. IEEE Access. 2017 Dec 19;6:22764-77.

[151] Dang VT, Huong TT, Thanh NH, Nam PN, Thanh NN, Marshall A. Sdn-based syn proxy—a solution to enhance performance of attack mitigation under tcp syn flood. The Computer Journal. 2019 Apr 1;62(4):518-34.

[152] Sinha M. SynFloWatch: A Detection System against TCP-SYN based DDoS Attacks using Entropy in Hybrid SDN. InProceedings of the 25th International Conference on Distributed Computing and Networking 2024 Jan 4 (pp. 359-364).

[153] Long LZ, Selvarajah V. An alternative DOS attack on windows. InAIP Conference Proceedings 2024 Jan 25 (Vol. 2802, No. 1). AIP Publishing.

[154] Mutlaq KA, Nyangaresi VO, Omar MA, Abduljabbar ZA. Symmetric Key Based Scheme for Verification Token Generation in Internet of Things Communication Environment. InApplied Cryptography in Computer and Communications: Second EAI International Conference, AC3 2022, Virtual Event, May 14-15, 2022, Proceedings 2022 Oct 6 (pp. 46-64). Cham: Springer Nature Switzerland.

[155] Tandon R. A survey of distributed denial of service attacks and defenses. arXiv preprint arXiv:2008.01345. 2020 Aug 4.

[156] Qian Z, Mao ZM, Xie Y. Collaborative TCP sequence number inference attack: how to crack sequence number under a second. InProceedings of the 2012 ACM conference on Computer and communications security 2012 Oct 16 (pp. 593-604).

[157] Yang G. Introduction to TCP/IP network attacks. Secure Systems Lab. 1997 Nov.

[158] Ahmadi S. Challenges and Solutions in Network Security for Serverless Computing. International Journal of Current Science Research and Review. 2024 Jan 11;7(01):218-29.

[159] Ahmed SF, Alam MS, Afrin S, Rafa SJ, Rafa N, Gandomi AH. Insights into Internet of Medical Things (IoMT): Data fusion, security issues and potential solutions. Information Fusion. 2024 Feb 1;102:102060.

[160] 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.

[161] Harris B, Hunt R. TCP/IP security threats and attack methods. Computer communications. 1999 Jun 25;22(10):885-97.

[162] Yihunie F, Abdelfattah E, Odeh A. Analysis of ping of death DoS and DDoS attacks. In2018 IEEE Long Island Systems, Applications and Technology Conference (LISAT) 2018 May 4 (pp. 1-4). IEEE.

[163] Yudhana A, Riadi I, Suharti S. Distributed Denial of Service (DDoS) Analysis on Virtual Network and Real Network Traffic. Journal of informatics and telecommunication engineering. 2021 Jul 16;5(1):112-21.

[164] Pande S, Khamparia A, Gupta D, Thanh DN. DDOS detection using machine learning technique. InRecent Studies on Computational Intelligence: Doctoral Symposium on Computational Intelligence (DoSCI 2020) 2021 (pp. 59-68). Springer Singapore.

[165] Walad SI, Zarlis M, Efendi MI. Analysis of denial of service attack on web security systems. InJournal of Physics: Conference Series 2021 Mar 1 (Vol. 1811, No. 1, p. 012127). IOP Publishing.

[166] 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.

[167] Abdollahi A, Fathi M. An intrusion detection system on ping of death attacks in IoT networks. Wireless Personal Communications. 2020 Jun;112(4):2057-70.

[168] Chen Y, Hwang K, Kwok YK. Collaborative defense against periodic shrew DDoS attacks in frequency domain. ACM transactions on information and system security. 2005 May;30.

[169] Chang CW, Lee S, Lin B, Wang J. The taming of the shrew: Mitigating low-rate TCP-targeted attack. IEEE Transactions on Network and Service Management. 2010 Feb 17;7(1):1-3.

[170] Shanmugam T, Chellappan C. A DiffServ Policy based approach for improved Shrew Attack Protection. In2011 Third International Conference on Advanced Computing 2011 Dec 14 (pp. 34-40). IEEE.

[171] Abu-Alhaj MM, Hussein AH, Kharma Q, Shambour Q. Multi-indicator Active Queue Management Method. Comput. Syst. Sci. Eng.. 2021 Jan 1;38(2):251-63.

[172] Baklizi M. Weight queue dynamic active queue management algorithm. Symmetry. 2020 Dec 14;12(12):2077.

[173] Oran S, Koçak A, Alkan M. Security Review and Performance Analysis of QUIC and TCP Protocols. In2022 15th International Conference on Information Security and Cryptography (ISCTURKEY) 2022 Oct 19 (pp. 25-30). IEEE.

[174] 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.

[175] Dik D, Berger MS. Open-RAN fronthaul transport security architecture and implementation. IEEE Access. 2023 May 8.

[176] Roy Chowdhury R, Che Idris A, Abas PE. Internet of things device classification using transport and network layers communication traffic traces. International Journal of Computing and Digital Systems. 2022 Aug 6;12(1):545-55.

[177] Saddheer M, Ahmad W, Nadeem M, Zahra SW, Arshad A, Riaz S. A Decrease in the Encryption Latency Utilizing Transport Layer Protocols for Software Defined Networks.

[178] Aslan FY, Aslan B. Comparison of IoT Protocols with OSI and TCP/IP Architecture. International Journal of Engineering Research and Development. 2023 Jan 1;15(1):333-43.

[179] Shilpa V, Vidya A, Pattar S. MQTT based secure transport layer communication for mutual authentication in IoT network. Global Transitions Proceedings. 2022 Jun 1;3(1):60-6.

[180] Abduljabbar ZA, Abduljaleel IQ, Ma J, Al Sibahee MA, Nyangaresi VO, Honi DG, Abdulsada AI, Jiao X. Provably secure and fast color image encryption algorithm based on s-boxes and hyperchaotic map. IEEE Access. 2022 Feb 11;10:26257-70.

[181] de Carné de Carnavalet X, van Oorschot PC. A Survey and Analysis of TLS Interception Mechanisms and Motivations: Exploring how end-to-end TLS is made "end-to-me" for web traffic. ACM Computing Surveys. 2023 Jul 13;55(13s):1-40.

[182] Waked L, Mannan M, Youssef A. The sorry state of TLS security in enterprise interception appliances. Digital Threats: Research and Practice. 2020 May 29;1(2):1-26.

[183] Arfaoui G, Bultel X, Fouque PA, Nedelcu A, Onete C. The privacy of the TLS 1.3 protocol. Cryptology ePrint Archive. 2019.

[184] Alwazzeh M, Karaman S, Shamma MN. Man in The Middle Attacks Against SSL/TLS: Mitigation and Defeat. J. Cyber Secur. Mobil.. 2020 Jul;9(3):449-68.

[185] Frolov S, Wustrow E. The use of TLS in Censorship Circumvention. InNDSS 2019 Feb.

[186] 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.

[187] Lenard T, Bolboaca R. A statefull firewall and intrusion detection system enforced with secure logging for controller area network. InProceedings of the 2021 European Interdisciplinary Cybersecurity Conference 2021 Nov 10 (pp. 39-45).

[188] Volkova A, Niedermeier M, Basmadjian R, de Meer H. Security challenges in control network protocols: A survey. IEEE Communications Surveys & Tutorials. 2018 Sep 26;21(1):619-39.

[189] Tournier J, Lesueur F, Le Mouël F, Guyon L, Ben-Hassine H. A survey of IoT protocols and their security issues through the lens of a generic IoT stack. Internet of Things. 2021 Dec 1;16:100264.

[190] Gratzer F, Gallenmüller S, Scheitle Q. Quic-quick udp internet connections. Future Internet and Innovative Internet Technologies and Mobile Communications. 2016 Sep.

[191] Singh S, Sulthana R, Shewale T, Chamola V, Benslimane A, Sikdar B. Machine-learning-assisted security and privacy provisioning for edge computing: A survey. IEEE Internet of Things Journal. 2021 Jul 19;9(1):236-60.

[192] 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.

[193] Borges de Oliveira F, Borges de Oliveira F. Selected Privacy-Preserving Protocols. On Privacy-Preserving Protocols for Smart Metering Systems: Security and Privacy in Smart Grids. 2017:61-100.

[194] Saha TK, Koshiba T. An efficient privacy-preserving comparison protocol. InInternational Conference on Network-Based Information Systems 2017 Aug 24 (pp. 553-565). Cham: Springer International Publishing.

[195] Jiang Q, Deng K, Zhang L, Liu C. A privacy-preserving protocol for utility-based routing in DTNs. Information. 2019 Apr 8;10(4):128.

[196] Liu Z, Wang L, Bao H, Cao Z, Zhou L, Liu Z. Efficient and Privacy-Preserving Cloud-Assisted Two-Party Computation Scheme in Heterogeneous Networks. IEEE Transactions on Industrial Informatics. 2024 Mar 7.

[197] Miao J, Wang Z, Wu Z, Ning X, Tiwari P. A blockchain-enabled privacy-preserving authentication management protocol for Internet of Medical Things. Expert Systems with Applications. 2024 Mar 1;237:121329.

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

[199] Wang J, Liu L, Lyu S, Wang Z, Zheng M, Lin F, Chen Z, Yin L, Wu X, Ling C. Quantum-safe cryptography: crossroads of coding theory and cryptography. Science China Information Sciences. 2022 Jan;65(1):111301.

[200] Wu C, Lu H, Chen Y, Qin L. Cross-Layer Optimization for Statistical QoS Provision in C-RAN with Finite-Length Coding. IEEE Transactions on Communications. 2024 Feb 26.

[201] Shah SW, Mian AN, Aijaz A, Qadir J, Crowcroft J. Energy-efficient mac for cellular IoT: state-of-the-art, challenges, and standardization. IEEE Transactions on Green Communications and Networking. 2021 Feb 25;5(2):587-99.

[202] Lee E, Seo YD, Oh SR, Kim YG. A Survey on Standards for Interoperability and Security in the Internet of Things. IEEE Communications Surveys & Tutorials. 2021 Mar 19;23(2):1020-47.

[203] Guo J, Wang L, Zhou W, Wei C. Powering green digitalization: evidence from 5G network infrastructure in China. Resources, Conservation and Recycling. 2022 Jul 1;182:106286.

[204] 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.

[205] Deng S, Zhao H, Fang W, Yin J, Dustdar S, Zomaya AY. Edge intelligence: The confluence of edge computing and artificial intelligence. IEEE Internet of Things Journal. 2020 Apr 1;7(8):7457-69.

[206] Andrade-Hoz J, Wang Q, Alcaraz-Calero JM. Infrastructure-Wide and Intent-Based Networking Dataset for 5G-and-beyond AI-Driven Autonomous Networks. Sensors. 2024 Jan 25;24(3):783.

[207] Soltani R, Zaman M, Joshi R, Sampalli S. Distributed ledger technologies and their applications: A review. Applied Sciences. 2022 Aug 6;12(15):7898.

[208] García CR, Rommel S, Takarabt S, Olmos JJ, Guilley S, Nguyen P, Monroy IT. Quantum-resistant Transport Layer Security. Computer Communications. 2024 Jan 1;213:345-58.

[209] Papavassiliou S. Software defined networking (SDN) and network function virtualization (NFV). Future Internet. 2020 Jan 2;12(1):7.