MEST Journal

FIREWALL-BASED DEFENSE STRATEGIES AGAINST MAN-IN-THE-MIDDLE ATTACKS

Zoran Cekerevac
Independent Researcher, Belgrade, Serbia
https://orcid.org/0000-0003-2972-2472

JEL Category: **C88, D83, K24, L86, O33**

Abstract

Firewalls serve as the first line of defense against Man-in-the-Middle (MITM) attacks, which compromise the confidentiality, integrity, and authenticity of digital communication. This paper presents a structured taxonomy of core MITM techniques, including ARP poisoning, DNS spoofing, HTTPS degradation (also known as SSL stripping), and session hijacking, as well as specialized variants targeting cloud services, web browsers, mobile applications, and IoT devices. Particular attention is given to vulnerabilities in VPN infrastructures, where centralized traffic decryption creates high-value targets, and to weaknesses in IoT ecosystems stemming from unvalidated certificates and outdated factory configurations. The analytical-comparative methodology integrates a literature review, statistical assessment of the economic impact of MITM incidents, and a practical demonstration of advanced firewall capabilities using Linux iptables/nftables configuration. The paper outlines both fundamental and advanced features of modern firewall solutions, including ACL rules, stateful inspection, application-layer filtering, DNS filtering, TLS inspection, and integration with IDS/IPS systems. Illustrative examples from widely used applications highlight the strengths and limitations of these measures. The findings emphasize that while firewalls are essential, they are not sufficient on their own. Effective protection requires a multilayered architecture that combines DNS encryption, strict TLS certificate validation, anomaly detection, and continuous user education to significantly reduce the risks and economic consequences of MITM attacks in contemporary digital networks.

Keywords: *firewall, MITM attacks, ARP poisoning, DNS spoofing, TLS inspection, IDS/IPS, VPN security, IoT vulnerabilities*

1 INTRODUCTION

1.1 Fundamentals of Hacking and Hacker Motivation

Hacking encompasses all unauthorized methods of interacting with information systems, where actors—commonly referred to as hackers—seek to access, modify, or steal data beyond the permissions granted by system owners. Hackers are ethically categorized into three groups: black-hat (malicious), white-hat (ethical), and gray-hat (ambivalent). While they may share tools and techniques, their motivations and objectives differ significantly. Skill levels range from advanced malware developers to so-called “script kiddies” who exploit known vulnerabilities without deep technical understanding.

Address of the author:
Zoran Čekerevac
zoran@cekerevac.eu

Motivations for hacking include data harvesting, impersonation for fraudulent activities such as DDoS attacks, destructive intent aimed at system disruption, and personal thrill-seeking or challenge-driven behavior. The legal framework surrounding hacking is complex, involving issues such as data ownership versus system ownership, privacy concerns, and ambiguous boundaries of lawful online conduct. Hackers are subject to legal consequences for unauthorized actions.

Hacking techniques span from physical device theft to sophisticated network-based attacks that exploit open ports, backdoors, and social engineering tactics such as phishing[1] and credential spoofing. Even software "Easter eggs"[2] left by developers can be repurposed for malicious use. One of the most impactful forms of hacking is the Man-in-the-Middle (MITM) attack, which is the central focus of this paper and is examined in detail in Section 3.

Successful hacking requires deliberate planning, considering the attacker's goals, profitability, chosen methods (mass versus targeted attacks), and potential consequences. Effective attacks demand substantial effort and are rarely instantaneous.

1.2 Financial Motivation

Cyberattacks result in substantial financial losses, including direct theft, remediation costs, and ongoing security expenditures. As early as 2011, hackers reportedly generated over $12.5 billion in illicit revenue, with corporations suffering significant financial damage. Despite advancements in cybersecurity, the scale of losses has continued to grow.

For example, the average cost of a data breach in 2024 reached $4.88 million per organization, marking a 10% increase compared to $4.45 million in 2023. Interestingly, this figure declined by 9% in 2025 to $4.44 million, which may suggest improved defensive measures or a shift in attacker focus on smaller organizations. (IBM, 2025).

According to Verizon's 2025 Data Breach Investigations Report (2025), several key findings stand out:

- 30% of security breaches involved third parties, doubling from the previous year, with root causes including system vulnerabilities and operational disruptions.
- The number of attackers exploiting vulnerabilities for initial access rose by 34% year-over-year.
- Organizations remediated 54% of perimeter vulnerabilities, while nearly half remained unresolved.
- 44% of analyzed incidents involved ransomware, representing a significant increase from prior reports.

The emergence of AI tools such as ChatGPT in late 2022 triggered an explosion of phishing campaigns. Within six months, phishing volume increased more than fortyfold compared to previous periods (SOCRadar, 2024). Before AI adoption, phishing emails were often grammatically flawed and easily detectable. ChatGPT enabled attackers to generate grammatically correct, stylistically convincing messages that appeared to originate from colleagues, banks, or IT support. This led to higher success rates. Moreover, AI can produce hundreds of phishing message variants per second, and attackers now deploy scripts that combine ChatGPT with automated email delivery systems.

In 2021, the average cost of a cyber incident exceeded €670,000 per hour of active attack duration, while the number of compromised accounts surpassed one billion (AAG, 2025).

According to CVEdetails.com, which tracks publicly disclosed CVE identifiers, over 40,000 vulnerabilities were reported in 2024, and more than 30,000 were registered in the first seven months of 2025 (CVEdetails, 2025). Based on

[1] Phishing refers to an attempt to steal sensitive information—such as usernames, passwords, credit card numbers, or bank account details—for misuse or resale. Disguised as a trusted source with an enticing request, the attacker lures victims into deception, much like a fisherman uses bait to catch a fish.

[2] Software Easter eggs are hidden features, messages, or content intentionally embedded by developers, not part of the official functionality. They are often humorous, nostalgic, or playful, and typically require a specific sequence of actions to be revealed. Examples include typing "do a barrel roll" in Google Search, which rotates the page, or entering "about:robots" in Mozilla Firefox to display a whimsical message about robots.

analysis using tools like CVEmap and data from the GitHub community, it is estimated that 35–45% of these vulnerabilities had publicly available proof-of-concept (PoC) exploit code, making them technically accessible for exploitation. These vulnerabilities were or could have been leveraged in various attack types, including ransomware, phishing, supply chain compromise, privilege escalation, remote code execution, and zero-day exploitation.

The Hackmanac Cyber Threat Report 2024 estimated the average financial impact of a cyberattack at approximately $5 million, encompassing ransom payments, technical remediation, reputational damage, and business interruption (Hackmanac, 2024). However, this average varies widely depending on the target: while large organizations may suffer multimillion-dollar losses, individuals and small businesses often experience lower—but still significant—damage.

According to the NETSCOUT Threat Intelligence Report, the first half of 2024 saw an average of 41,000 DDoS attacks per day, totaling approximately 7.5 million attacks—a 30% increase compared to the same period in the previous year (NETSCOUT, 2024).

2 METHODOLOGY

In alignment with the research objectives, the following research questions were formulated:

RQ1: Is the firewall a viable tool for mitigating MITM attacks?

RQ2: If so, how can a properly configured firewall, in combination with a multilayered security architecture, effectively reduce the risk of MITM attacks in modern digital environments?

The study employed an analytical-comparative methodology, combining theoretical analysis, case studies, and technical demonstrations. The methodological framework was structured around the following components:

- Analysis of MITM attack techniques and their variants
- Comparison of traditional and advanced firewall functionalities
- Case studies of MITM scenarios in the context of Viber and WhatsApp applications
- Comparative assessment of VPN[3] and Tor[4] infrastructures with respect to security resilience
- Statistical overview of the financial impact of MITM incidents
- Demonstration of iptables rules in a Linux environment as a practical configuration example

The literature review encompassed academic databases such as Google Scholar, IEEE Xplore, SpringerLink, and MDPI, supplemented by specialized technical sources focused on firewall configuration. Due to the complexity of the topic, artificial intelligence tools were employed to optimize query formulation, abstract screening, and preliminary selection of relevant publications.

Inclusion criteria for sources were based on thematic relevance, methodological rigor, and publication date. Most of the analyzed works were published within the last five years, alongside prior publications by the author in the field of MITM attacks.

The collected material underwent critical analysis in accordance with academic standards for source validation, reliability assessment, and relevance to the defined research problem. The information was organized through thematic analysis, which identified key security domains, recurring patterns, and a conceptual framework that integrates existing knowledge.

For clarity and comparative purposes, selected findings were presented in tabular format. Based on the conducted analysis, gaps in the existing literature were identified, serving as a foundation for the formulation of conclusions and recommendations.

[3] A VPN (Virtual Private Network) is a technology that enables secure, encrypted communication between a user and a remote network over the public internet. By using a VPN, all network traffic is routed through a protected channel, ensuring data confidentiality, integrity, and anonymity

[4] Tor (The Onion Router) is a decentralized network and software bundle that enables anonymous communication over the internet. It uses multi-layered encryption and routes traffic through a series of randomly selected relay nodes, effectively concealing the user's identity and location. (Dingledine, et al., 2004)

3 MAN-IN-THE-MIDDLE ATTACKS

3.1 Introduction to MITM Attacks

MITM attacks represent a distinct class of cyber intrusions in which an attacker covertly positions themselves between two communicating parties with the intent to intercept, redirect, or alter transmitted data. These attacks pose a direct threat to the confidentiality, integrity, and authenticity of digital communication—affecting both corporate infrastructures and end-user environments.

Common vectors for MITM scenarios include unsecured Wi-Fi networks, DNS record manipulation (DNS spoofing, also known as DNS cache poisoning), removal of encryption layers in TLS/SSL protocols (SSL stripping[5]), and session hijacking. With the proliferation of cloud technologies, the Internet of Things (IoT), and Bring Your Own Technology (BYOT[6]) practices, MITM attacks have become increasingly prevalent. Adversaries exploit weaknesses in encryption, authentication mechanisms, and network protocols to gain unauthorized access and manipulate data flows.

3.2 Technological Foundations of MITM Attacks

MITM attacks have existed long before the advent of computers and can be likened to a malicious postal worker intercepting letters between two parties. Modern MITM attacks rely on sophisticated techniques that exploit vulnerabilities in security protocols. Particularly problematic are implementations of SSL/TLS protocols, where administrative complexity often results in one-way authentication—leaving room for attackers to impersonate legitimate entities.

In a typical MITM scenario, the attacker intercepts communication, inserts themselves as an intermediary between the communicating parties, and manipulates message content. For example, they may alter invoice payment details to redirect funds to their own account.

The most common MITM techniques include (OWASP, 2025):

- ARP cache poisoning[7]
- DNS spoofing
- Session hijacking, including side-jacking, evil twin attacks, and packet sniffing
- SSL session hijacking

The technological basis of MITM attacks is extensively discussed in Čekerevac et al. (2017a). Therefore, this paper presents only a schematic flow of a representative attack, highlighting the directional flow of information. Figure 1 illustrates how an attacker, positioned between two victims, intercepts and modifies communication, thereby compromising the integrity of transmitted data.

Figure 1 illustrates a typical MITM attack scenario:

- **Victim A** initiates communication by sending a “Hello” message, which the attacker intercepts and forwards to **Victim B** without modification, creating the illusion of a direct connection.
- **Victim B** responds by sending their public key (“Public key B”), which the attacker intercepts and replaces with their own key (“Public key MITM”) before relaying it back to **Victim A**.
- When **Victim A** sends an invoice (“Invoice A”), the attacker modifies the content and forwards the altered version (“Changed invoice”) to **Victim B**.
- Ultimately, **Victim B** completes a payment to the wrong address (“Money paid to the wrong address”), believing they are communicating with a legitimate partner.

This schematic highlights the core stages of an MITM attack: interception, manipulation, and exploitation of communication. The attacker seamlessly integrates into the communication channel, leaving both parties under the false impression that they are interacting directly.

[5] SSL stripping is a technique in which an attacker intercepts HTTPS requests and redirects them to the unencrypted HTTP version of the target site. By removing the encryption layer, the attacker gains access to plaintext data, enabling the theft of sensitive information such as login credentials, session tokens, and personal details.

[6] Bring Your Own Technology (BYOT) je strategija ili praksa koja omogućava zaposlenima, studentima ili saradnicima da koriste svoje lične uređaje i tehnologije (laptopove, pametne telefone, aplikacije, cloud servise itd.) za pristup organizacionim resursima, mrežama i podacima.

[7] ARP cache poisoning, also known as ARP spoofing, is a type of cyberattack that targets vulnerabilities in the Address Resolution Protocol (ARP)—a protocol that maps IP addresses to MAC addresses within a local area network (LAN). Commonly used tools for this purpose include Ettercap, Scapy, Cain & Abel, Bettercap, and MITMf

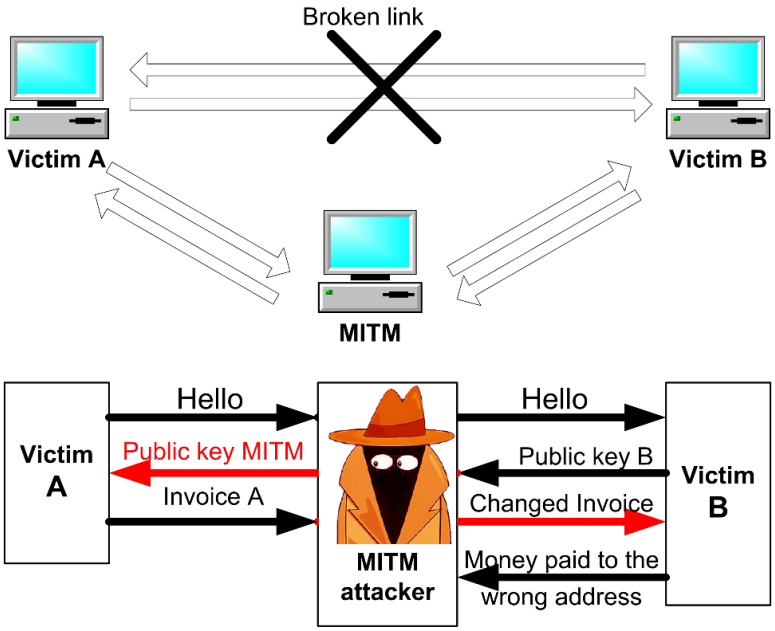


Figure 1. Schematic representation of an MITM attack: interception and manipulation of messages between two victims.

Izvor: Cekerevac et al. (2017a)

An illustrative example of such exploitation involves intercepting FTP credentials using tools like dsniff, which allow attackers to capture usernames and passwords transmitted in plaintext—opening the door to further network compromise.

3.3 Evolution of MITM Attacks

With the advancement of technology, MITM attacks have evolved into several specialized forms:

- *MIT-cloud (MITC): Exploitation of session tokens in cloud services to gain unauthorized access.*
- *MIT-mobile (MITMO): Interception of mobile authentication codes (mTAN) transmitted via SMS.*
- *MIT-app (MITA): Injection of self-signed certificates to intercept data exchanged through mobile or desktop applications.*
- *MIT-IoT: Targeting IoT devices that fail to validate SSL certificates, enabling credential theft and unauthorized control.*

Note: Although similar in name, Man-in-the-Browser (MITB) attacks are not technically classified as MITM attacks. MITB exploits occur locally within the user's web browser via malware, without intercepting network traffic between the client and server.

3.4 MITM Attacks in the Context of IoT Devices

Internet of Things (IoT) devices are particularly vulnerable to MITM attacks due to limited computational resources, lack of regular updates, and weak encryption protocols. These attacks are typically executed locally over Ethernet or Wi-Fi networks, leveraging ARP poisoning, DNS manipulation, and HTTPS interception through self-signed certificates or tools such as SSLstrip.

Many smart devices still fail to validate TLS/SSL certificates, allowing attackers to intercept credentials and compromise connections using tools like Ettercap, Evilgrade, dsniff, and Cain & Abel (Vahab, 2025). According to OWASP, “Insecure Data Transfer and Storage” remains one of the most widespread IoT vulnerabilities, as a significant number of devices do not verify certificate trust chains and often rely on outdated cryptographic libraries.

Bluetooth Low Energy (BLE), commonly found in smart locks, thermostats, and security cameras, exhibits a high rate of vulnerability. Hlapisi (2023) reports that 70–80% of tested BLE models are susceptible to cloning, passive data interception, and unauthorized device takeover. These recent findings confirm and expand upon earlier observations by Spring (2016), underscoring the urgent need for stricter certificate validation

mechanisms and regular firmware updates (Watlecorp, 2025; Hlapisi, 2023).

Distributed Denial-of-Service (DDoS) and related Denial-of-Service (DoS) attacks accounted for approximately 60–64% of all attacks targeting IoT devices in 2016, according to reports by McAfee and OWASP (McAfee, 2016). Billions of IoT devices remain continuously connected to poorly monitored networks, making them ideal targets for botnet recruitment, spam distribution, and credential theft. Notable examples include attacks on smart hubs and refrigerators that transmit unencrypted data.

IoT devices are frequently shipped with insecure factory settings, default passwords, open interfaces, outdated firmware, and a lack of update mechanisms, which make them highly susceptible to exploitation. Connected vehicles are also vulnerable, as demonstrated in the 2015 Jeep Cherokee incident, where remote hacking led to the recall of 1.4 million cars (Cekerevac et al., 2017).

3.5 VPN Infrastructure as a Potential MITM Attack Vector

Public hotspot networks—especially those that are open and unsecured—are classic environments for executing MITM attacks. However, due to their limited range and physical accessibility, such attacks typically target local or narrowly defined user groups. In contrast, VPN infrastructure, which serves as a centralized conduit for encrypted traffic from many users, presents a far broader surface for potential compromise.

Although VPN services are commonly perceived as privacy-enhancing security mechanisms, their architecture inherently requires that all traffic be decrypted on the server side before being forwarded to its destination. If a VPN provider were malicious or compromised, it could intercept, modify, and analyze user traffic—creating conditions for a sophisticated MITM attack. Particularly vulnerable are unencrypted protocols, unvalidated certificates, and unchecked DNS requests.

Given the level of trust users place in VPN services, it is essential to select solutions that offer transparent policies, open-source codebases, independent security audits, and technical safeguards that minimize the risk of abuse—whether by the provider itself or in the event of a breach. Security is further enhanced when VPN providers undergo external verification by independent organizations. One such example is the Mobile Application Security Assessment (MASA) conducted on the Mullvad VPN service (Mullvad, 2025).

In the context of communication security, it is critical to distinguish between two foundational concepts:

- *Infrastructure trust*: VPN services operate on the assumption that the provider is trustworthy, does not log user activity, refrains from third-party cooperation, and is technically capable of securing user traffic. However, users have no direct control over these assurances—trust is external and often unverifiable.
- T*echnical assurance of security*: The Tor network, although slower, is based on a decentralized model with multilayered encryption (onion routing), where no single node knows the complete transmission path. Security is not dependent on trust in individual entities but on the architecture itself. While the exit node may be a point of surveillance, preceding layers protect the user's identity.

A comparative overview of VPN and Tor architectures is presented in Table 1.

Table 1. VPN vs. Tor — Security Considerations

Aspect	VPN	Tor
Speed	Faster; suitable for streaming and daily tasks	Slower due to multilayered routing
Privacy	Depends on the provider	Embedded in the system architecture
Resistance to MITM Attack	Vulnerable if the provider is compromised or cooperates with third parties	Stronger resistance; exit node remains a weak point
Visibility and Control Over Communication*	Limited user autonomy	Greater anonymity, but less granular control
Suitability for Daily Use	High	Lower, but valuable for specific use cases

Source: Author

* *Note*: In this context, “visibility and control over communication” refers to the user’s ability to influence security parameters, understand network architecture, and manage their own level of privacy and anonymity. With VPNs, this autonomy is constrained by trust in the provider, whereas Tor offers greater technical independence.

3.6 Strategies in MITM Attacks: Adversaries vs. Victims

In the context of MITM attacks, strategic thinking is not exclusive to defenders—attackers also develop sophisticated approaches to evade detection and maximize effectiveness. Understanding the tactics employed by both sides enables a more precise definition of security requirements and facilitates more effective defense mechanisms.

3.6.1 Adversarial Self-Preservation Strategies

Attackers typically adopt a range of measures to minimize the likelihood of exposure:

- Operating remotely and frequently changing physical locations
- Utilizing publicly accessible or disposable devices
- Conducting financial transactions via cash or prepaid instruments to reduce traceability
- Employing anonymization tools (e.g., VPN, Tor) and automated scripts to hinder attribution.

3.6.2 Defensive Strategies for Victims

While complete elimination of MITM attacks remains challenging, risk can be significantly reduced through the implementation of the following measures:

- Designing network architectures with security as a foundational principle
- Deploying security-oriented network topologies
- Regularly updating operating systems and software
- Using firewalls and strong encryption mechanisms (e.g., SSL/TLS certificates)
- Implementing static ARP entries to prevent ARP poisoning
- Avoiding connections to unsecured Wi-Fi networks and using tools such as HTTPS Everywhere
- Applying DNSSEC and intrusion detection systems to mitigate DNS spoofing attacks
- Promoting digital literacy and cultivating a culture of cybersecurity awareness among users

Organizations, particularly smaller ones with limited resources, often need to reassess their security practices or engage external protection services. Key challenges include threat awareness and timely detection of attacks.

The use of public key cryptography (PKC) and digital certificates issued by trusted Certificate Authorities (CAs) is essential for reliable device identification and secure communication. However, compromise of root keys, the highest-level trust anchors, can jeopardize the entire system. This underscores the importance of the root of trust[8] concept within trusted computing modules. Although cryptographic methods provide foundational protection, their effectiveness is significantly enhanced when combined with properly configured firewall systems, which will be examined in detail in the following section.

Users are advised to disable automatic network connections, avoid opening suspicious links and attachments, and refrain from performing device modifications such as *jailbreaking* or *rooting*, to reduce the risk of MITM attacks.

Given the increasing interconnectivity of devices and the complexity of modern network environments, MITM attacks represent not only a technical challenge but also a significant security and economic risk—particularly in the context of digital business transformation.

Due to their frequency and ability to compromise critical communication flows, MITM attacks occupy a prominent position among cyber threats. The following subsection presents their financial impact through statistical indicators collected over the past five years.

[8] Roots of trust refer to foundational components within hardware or software infrastructures that are considered inherently reliable. In the context of a Trusted Platform Module (TPM), these include cryptographic keys and mechanisms that initiate core security operations such as firmware verification, data encryption, and system authentication.

3.7 Economic Impact Assessment of MITM Attacks

MITM attacks represent a technically sophisticated form of cyber threat, capable of covertly intercepting communication, stealing credentials, and manipulating sessions and transactions. Their destructiveness extends beyond direct financial damage, undermining systemic trust in digital infrastructures.

Within the broader spectrum of cyberattacks, MITM occupies a prominent position—both in terms of frequency and economic impact. According to available data, MITM accounts for approximately 19% of successful online attacks, with an estimated annual cost of $2.4 billion (Astra Security, 2023). In the domain of Wi-Fi exploitation, MITM techniques contribute to as much as 35% of incidents, making them one of the most prevalent attack vectors in wireless environments. Additionally, 50% of MITM incidents result in credential theft, with over one million passwords compromised monthly, highlighting a serious risk to user identities and enterprise security systems.

Manufacturing enterprises are particularly vulnerable due to extensive use of IoT devices, automated systems, and often inadequately secured network configurations. MITM attacks are increasingly combined with automation and artificial intelligence, enhancing their efficiency and reducing the need for direct attacker involvement.

For a more comprehensive overview, Table 2 presents a comparative analysis of MITM and other dominant cyberattack types observed between 2021 and 2025.

Table 2. Types of Cyberattacks: Characteristics, Prevalence, and Financial Impact (2021–2025)

Attack Type	Typical Target	Prevalence (2021-2025)	Annual Cost	Dominant Vector	Data Theft Share	Source(s)
MITM	Communication, passwords, sessions	High (19% of online attacks)	$2.4 billion	AI, phishing, Wi-Fi	50% of MITM incidents involve credential theft	(Wabuge, 2023)
Phishing	User data, credentials	Very high (3.4 billion phishing emails per day)	$3.1 billion (estimated)	Email, links	41% of incidents begin with phishing	(Palatty, 2025), (SSL Insights, 2025), (APWG, 2025)
Ransom-ware	Systems, databases	Declined from 66% in 2023 to 59% in 2024, rising again in late 2024	$20B (2021); $57B (2025); avg. $3.9M per incident	Encryption, extortion	Medium (60% involve data loss)	(Okoruwa & Chapman, 2025), (Morgan, 2025) (Threat Hunter Team, 2025)
DDoS	Service availability	Moderate, rapidly increasing (4× growth Q4 2022 vs Q4 2021)	$1.6 billion	Botnets	Low	(StormWall, 2025), (Smith, 2025)
Supply-chain	Software chains, update systems	Rising (431% increase)	Difficult to estimate	Compromised updates	Variable	(Morgan, 2023), (Snape, 2025)
SQL Injection	Databases	Moderate	Localized damage	Automated code	Low	(Jackson, 2024), (Citakovic, 2023)

Source: Author

Despite often being overlooked in public discourse, MITM attacks carry high strategic significance. Their ability to integrate with other vectors—such as phishing—and to target communication flows makes them particularly dangerous in the context of digital transformation and industrial automation.

Compared to ransomware attacks, which generate immediate financial damage, MITM attacks operate more quietly yet systemically undermining authentication, data integrity, and user trust. For this reason, it is essential to develop layered defense strategies, including end-to-end (E2E) encryption, network segmentation, anomaly detection, and user education.

4 FIREWALL

The implementation of robust network security measures forms the foundation of protection against Man-in-the-Middle (MITM) attacks, with the network firewall serving as the first line of defense between the user's internal network and external threats. To assess its functional value within MITM scenarios, first, it is essential to examine attack techniques, analyze the core and advanced capabilities of modern firewall solutions, and demonstrate the application of recommended security measures through case studies.

Firewalls based on traditional packet-level filtering—such as port, IP address, and protocol inspection—are often ineffective in detecting compromised sessions when attackers utilize permitted and encrypted communication channels. A standard Layer 3/Layer 4 (L3/L4)[9] firewall without TLS inspection enabled can observe only metadata (IP address, port, Server Name Indication[10]), but not the validity of certificates. The client within the application or operating system performs the certificate validation. For further details, see Section 4.2.2.

A properly configured firewall is not merely a nominal security measure—it represents the critical distinction between formal protection and actual system resilience. The following configuration strategies are designed to identify and block suspicious activities commonly associated with Man-in-the-Middle (MITM) attacks.:

1. Access Control Lists (ACLs)
   - Precisely define who can access which resources, under what conditions, and through which protocols.
   - Enforce a default-deny policy, blocking all traffic that is not explicitly permitted.
2. Stateful Inspection
   - The firewall analyzes not just individual packets but the entire session context.
   - Detects unauthorized attempts to inject packets into active sessions—a hallmark of MITM behavior.
3. Layer 7 Filtering (Application Layer)
   - Enables deep inspection of protocols such as HTTP, DNS, and FTP.
   - Blocks modified requests, unauthenticated responses, and anomalous traffic patterns.
4. Spoofing and ARP Manipulation Protection
   - Prevents IP address spoofing and ARP table tampering—common vectors in MITM scenarios.
   - Includes anti-spoofing rules and ARP monitoring mechanisms.
5. Logging and Alerting
   - A well-configured firewall logs access attempts, failed connections, and suspicious patterns.
   - Can trigger real-time alerts to administrators or forward events to a SIEM[11] system.
6. Regular Rule and Firmware Updates
   - Firewall rules must be continuously adapted to emerging threats.
   - Firmware updates ensure protection against known vulnerabilities and maintain system integrity.

4.1 Firewall Operation

Before analyzing firewall functionality, it is important to understand that a port, in the context of networking, represents a logical identifier that enables multiple applications on the same device to communicate with the network simultaneously. A port is not a physical entry point, but rather a software-defined channel that the operating system uses to route incoming data to the appropriate application. The application then utilizes system resources—such as the processor,

[9] L3 and L4 refer to the network and transport layers of the OSI model, respectively, and are commonly used as the basis for traffic filtering in standard firewalls. L3 filtering typically involves IP-based rules, while L4 filtering targets protocols such as TCP and UDP. (Kaspersky, 2025)

[10] SNI (Server Name Indication) is an extension of the TLS protocol that allows the client to specify the domain name it wishes to access during the TLS handshake. This enables servers to present the appropriate certificate for the requested domain, particularly in environments hosting multiple domains on a single IP address

[11] SIEM (Security Information and Event Management) systems are centralized platforms for managing security-related information and events within IT environments. Their primary function is to collect, analyze, and correlate data from various sources—such as network devices, servers, applications, and user accounts—to detect threats, anomalies, and security incidents in a timely manner.

memory, and other components—to process the received data.

The data handling process unfolds as follows:

- A network packet arrives at the network interface (e.g., Ethernet card).
- The packet is forwarded to the operating system's TCP/IP stack.
- The operating system uses the transport layer to identify the destination port and delivers the data to the application registered for that port.
- The application processes incoming data and, when necessary, utilizes the processor to execute tasks.

To receive data, the application must be actively running, configured for network communication, and authorized by the operating system. During this process, the firewall may either block or allow access to the designated port based on predefined security rules. Once registered to a specific port, the application begins accepting incoming connections.

Several characteristic scenarios are presented in Table 3.

Table 3. Examples of Port Usage by Applications

Application	Typical Port(s)	Persistent Listening?
Web server (e.g., Apache)	80 (HTTP), 443 (HTTPS)	Yes, if running
Email server (e.g., Postfix)	25 (SMTP)	Yes, if active
Web browser (e.g., Chrome)	Dynamic ports	No — initiates outbound connections only
Torrent client	6881–6889 (historically); dynamic ports now	Da, ako je pokrenut

Source: Author

Table 4. Port Classification in TCP/IP Networking

Port Type	Range	Purpose
Dobro poznati portovi	0–1023	Standard services (HTTP, HTTPS, FTP, SSH, DNS...)
Registered ports	1024–49151	Applications not part of the OS but widely recognized
Dynamic/private ports	49152–65535	Privremeni portovi za klijentske konekcije (ephemeral)

Source: Author

TCP/IP supports a total of 65,536 ports, which are divided into three categories as shown in Table 4.

In practice, only a relatively small number of ports are actively used.

- Servers rely on well-known ports to provide services, such as 80 (HTTP), 443 (HTTPS), 21 (FTP), 22 (SSH), 25 (SMTP), and 53 (DNS), among others (see Table 4).
- Clients use dynamic ports to initiate connections. For example, when a client opens a website, its device selects a port from the range 49152–65535 to communicate with the server on port 443.

On average, fewer than 100 ports are actively used on most systems, while the remaining ports remain closed or inactive.

In the best security practices, the following measures are recommended:

- Close all unused or unknown ports to minimize the attack surface.
- Log all access attempts, including failed connections and unauthorized probes.
- Deploy an Intrusion Detection System (IDS[12]) to monitor unusual activity patterns and detect potential threats in real time.

One may ask: *what happens when a packet arrives at a non-standard port, such as 54321?*

Port 54321 falls within the IANA-defined range for dynamic and private ports and is not standardized for any well-known service (e.g., 80 for HTTP or 443 for HTTPS). In practice, such ports may be used by custom applications, experimental services, or—more critically—by malicious software, including backdoor communication channels.

[12] IDS (Intrusion Detection System) and IPS (Intrusion Prevention System) are security mechanisms designed to identify, analyze, and respond to suspicious activity within network traffic.

Several outcomes may occur when a packet targets a non-standard port, such as 54321:

1. The port is closed (not permitted by firewall rules):
   - The firewall automatically drops the packet or responds with an ICMP[13] message *(reject)* indicating the port is unreachable.
   - The packet never reaches the operating system—the firewall intercepts the attempt.
2. The port is open, but no application is listening:
   - The operating system may ignore the request or return an error (e.g., TCP RST).
   - If the firewall operates in passive mode, an attacker may infer that the port is open, posing a potential security risk.
3. The port is open, and an application is active, but unprotected:
   - An attacker may attempt to exploit vulnerabilities in the application.
   - If the firewall does not inspect content, this may lead to buffer overflows, remote code execution, or even MITM attacks if the application uses insecure authentication mechanisms.

The firewall—or its operating system-integrated component—serves as the first layer of defense. Following this, the operating system maintains an internal table of active ports and the applications that have registered to use them. When a network packet arrives:

1. The operating system analyzes the port number specified in the packet.
2. It checks whether any application is actively listening on that port.
3. If a registered application is found, the data is forwarded accordingly.
4. If no application is listening, the packet is either discarded or a response is sent indicating that the port is unavailable.

A properly configured firewall can respond to incoming packets in several ways:

- *Drop:* Silently discards the packet before it reaches the operating system, without sending any response.
- *Ignore:* Passively disregards the packet, offering no acknowledgment or feedback to the sender.
- *Reject:* Actively denies the packet and returns an ICMP message indicating that the port or service is unavailable.

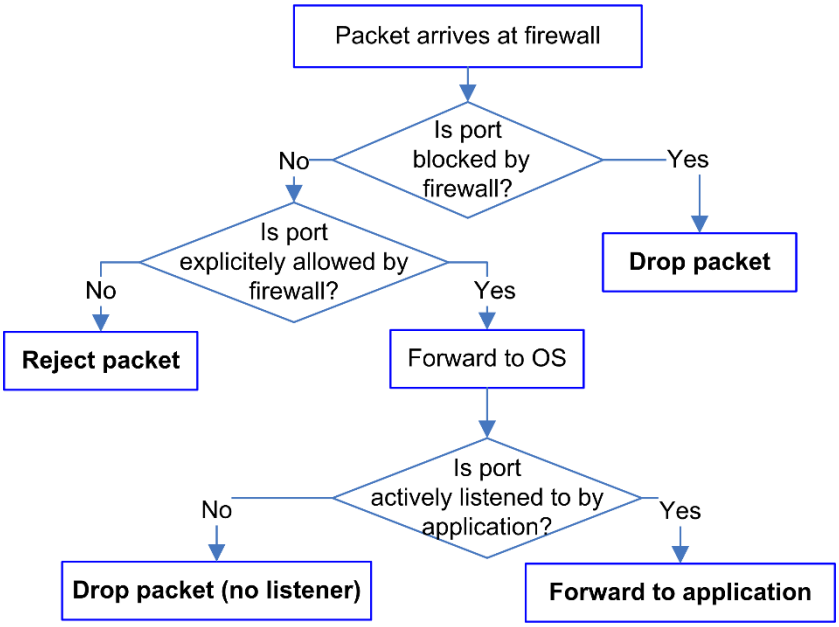


Figure 2. Packet Processing Flow

Source: Author

[13] An ICMP message (Internet Control Message Protocol) refers to a control-level network packet used to signal communication issues between devices on an IP network.

Practical Example

Figure 2 illustrates the decision-making algorithm for a network packet arriving at a port, e.g., 54321—highlighting the interaction between the firewall and the operating system (OS)

Two primary scenarios are possible at the firewall level:

- If the port is open on the firewall, but no application is listening:
  - The firewall allows the packet to pass through.
  - The OS discards the packet due to the absence of a registered application on that port.
  - The OS may respond with an ICMP "port unreachable" message (for UDP) or a TCP segment with the RST flag (for TCP)
- If the port is blocked by the firewall:
  - The packet never reaches the OS.
  - The firewall may silently drop the packet or explicitly reject it with a response.

The outcome depends on the configuration of both layers—the firewall acts as the first gatekeeper, while the operating system serves as the second.

The diagram visualizes the packet processing flow through the following decision-making stages:

- Firewall decisions: allow, block, or reject the packet
- Operating system decisions: based on whether an application is actively listening on the targeted port
- Outcome: drop, reject, or forward to the application.

A comparative overview of TCP and UDP protocol behavior in response to incoming packets is presented in Table 5.

Table 5. Comparative Analysis: TCP vs UDP

Characteristic	TCP (Transmission Control Protocol)	UDP (User Datagram Protocol)
Protocol type	Connection-oriented	Connectionless
Reliability	High — includes acknowledgment and retransmission	Low — no acknowledgment
Flow and error control	Yes	No
When no application is listening	OS sends TCP RST (reset)	OS sends ICMP Port Unreachable
Firewall behavior (drop)	Silently discards packet without response	Same — silent discard
Firewall behavior (reject)	May send TCP RST	May send ICMP message
Typical applications	Web servers, email, SSH, FTP	DNS, VoIP, video streaming
Speed	Slower due to control mechanisms	Faster, but less reliable
MITM vulnerability	Higher if used without encryption	Higher due to a lack of control mechanisms

Source: Author

In cases where an attacker attempts to probe port availability, system behavior depends on the protocol used:

- TCP: If a connection attempt is made to port 22 (SSH) and no application is actively listening, the operating system responds with a TCP RST (reset). This may signal to the attacker that the port exists but is inactive.
- UDP: If a packet is sent to port 53 (DNS) and no application is active, the operating system may respond with an ICMP Port Unreachable message—provided the firewall allows such responses.

Such responses may have security implications. TCP RST and ICMP messages can enable an attacker to map the network using tools as *nmap*. For this reason, many systems implement a firewall drop policy—silently discarding packets without any response—to reduce the likelihood of exposing active network services.

Depending on the requirements and security policy of the administrative team, two primary strategies are available for handling port behavior when no application is actively listening: silent dropping (*drop*) and active rejection (*reject*). These options are summarized in Table 6.

Table 6. Defining `iptables` Rules for Linux Firewall — Simulating TCP/UDP Behavior When No Application Is Listening

Policy	Linux Code Snippet
DROP (*silent discard*)	*# TCP packets to port 22 (SSH) are silently dropped* `iptables -A INPUT -p tcp --dport 22 -j DROP`
	# UDP packets to port 53 (DNS) are silently dropped `iptables -A INPUT -p udp --dport 53 -j DROP`
REJECT (*active denial*)	*# TCP packets to port 22 are rejected with TCP RST* `iptables -A INPUT -p tcp --dport 22 -j REJECT --reject-with tcp-reset`
	# UDP packets to port 53 are rejected with ICMP Port Unreachable `iptables -A INPUT -p udp --dport 53 -j REJECT --reject-with icmp-port-unreachable`

Source: Author

The listed commands are entered directly into `iptables`, the interface that manages the *netfilter* mechanism within the Linux kernel. Each command:

- Adds a rule to the INPUT chain (incoming traffic)
- Defines behavior for a specific port and protocol (TCP/UDP)
- Takes effect immediately and influences network traffic in real time

If the rules are not saved, they will be lost upon system reboot. The current configuration can be reviewed using the following command:
`iptables -L -n --line-numbers`

To ensure persistent firewall rules across system reboots, it is recommended to install the following package:
`sudo apt install iptables-persistent`

When using the iptables-persistent package, firewall rules are automatically stored in:

`/etc/iptables/rules.v4` (for IPv4)
`/etc/iptables/rules.v6` (for IPv6)

To enhance system security, it is recommended to set the default policy to DROP, which blocks all incoming and forwarded traffic unless explicitly allowed:

- `iptables -P INPUT DROP`
- `iptables -P FORWARD DROP`
- `iptables -P OUTPUT ACCEPT` (or `DROP`, depending on policy),

Next, rules for permitted services—such as SSH (port 22) and HTTP (port 80)—can be added:

- `iptables -A INPUT -p tcp --dport 22 -j ACCEPT`
- `iptables -A INPUT -p tcp --dport 80 -j ACCEPT`

Additionally, it is recommended to allow traffic from the loopback interface and already established connections:

- `iptables -A INPUT -i lo -j ACCEPT`
- `iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT`

When using `iptables` directly, rules are not saved automatically. To persist the current configuration, they must be manually saved:

`iptables-save > /etc/iptables/rules.v4`
(Debian/Ubuntu systems)

To manage firewall rules across system reboots, administrators may use different tools depending on the Linux distribution:

- iptables-persistent is commonly used in Debian and Ubuntu-based systems. It saves the current rule set and restores it automatically during startup.
- firewalld is the preferred solution in distributions such as Fedora, CentOS, and RHEL. It provides a dynamic firewall management interface and supports zone-based configurations.

The choice of tool depends on system architecture, administrative preferences, and compatibility with existing security policies.

4.1.1 Note on `nftables` in Modern Distributions

Although this text uses *iptables* to illustrate firewall rule definitions, modern Linux distributions—such as Fedora, Debian, and Ubuntu 22.04+—default to using nftables as the underlying backend. In many cases, *iptables* commands are internally translated into nftables rules through a compatibility layer

For example, blocking access to port 22 using *iptables* (Nickfetrat, 2024):

```
iptables -A INPUT -p tcp --dport 22 -j DROP
```

The equivalent *nftables* command would be:

```
nft add rule inet filter input tcp dport 22 drop
```

This *nftables* example uses the unified *inet* table. By leveraging the inet family, rules are automatically applied to both IPv4 and IPv6 protocols—eliminating the need for duplication.

To persist *nftables* rules across reboots, use:

```
nft list ruleset > /etc/nftables.conf
```

```
sudo systemctl enable nftables
```

On systems where *iptables* is still active, both interfaces may be used in parallel. However, transitioning to nftables is recommended due to its improved flexibility, performance, and native IPv6 support.

Regardless of whether iptables or nftables is used, proper firewall configuration remains a cornerstone of network security. By combining local and network-level rules, administrators can precisely control application access, minimize exposure to threats, and ensure stable system operation.

4.1.2 Note on Application Behavior During Software Installation

When installing new software, applications typically request permission to access the network. The operating system checks whether a local firewall is active. If so, the user is prompted with a dialog box asking: "*Allow this application to access the network?*"

If the user grants permission, the operating system automatically adds the corresponding rule to the local firewall, enabling the application to utilize network resources. The local firewall (e.g., Windows Firewall or *iptables*) governs the application's access to specific ports. The network firewall (e.g., on a router or gateway) may further filter traffic to external servers.

For example, applications such as WhatsApp and Viber use dynamic ports for outbound connections (typically in the range 50000–60000), but connect to predefined servers. It is important to note that the application does not "bypass" the firewall—it uses system APIs to request network access. If the firewall denies the connection, the application will be unable to establish communication.

4.2 How Firewalls Mitigate MITM Attacks?

Man-in-the-Middle (MITM) attacks often begin by exploiting vulnerabilities at the application layer, where an attacker targets the port used by a vulnerable application. If protection is insufficient, the attacker may:

- Execute malicious code
- Gain control over the application
- Indirectly access the processor and system resources

The firewall plays a critical role in preventing such scenarios. Through well-defined access policies, inspection of application-layer protocols, and integration with threat detection systems, the firewall can identify unauthorized access attempts and block them before the system is compromised.

The following section explores the firewall as the first line of defense against MITM attacks.

4.2.1 Firewall Defense Against MITM Attacks

A properly configured firewall can play a significant role in preventing Man-in-the-Middle (MITM) attacks. Its effectiveness depends on the direction of traffic and the point of potential compromise. In the context of MITM protection, two primary categories are distinguished:

- Inbound MITM: This type of attack involves unauthorized attempts to access the user's network from external sources, typically through open ports or vulnerable services. A firewall equipped with well-defined inbound traffic rules and active packet inspection can effectively block such connections.

A firewall with support for *stateful inspection* does not merely analyze individual packets—it monitors the entire session state, including IP addresses, ports, protocols, and communication flows. This enables dynamic decision-making regarding traffic legitimacy. The firewall can block unauthorized access attempts that do not belong to established sessions, including spoofed packets and injection attempts aimed at redirecting or manipulating the communication stream.

- Outbound MITM: In this scenario, an application on the user's device attempts to connect to a compromised external server. A firewall with DNS filtering, reputation-based lists, and TLS inspection capabilities can identify and block suspicious IP addresses, domains, or invalid certificates—preventing the establishment of harmful communication.
  In outbound MITM protection, stateful inspection allows the firewall to track the flow of outbound sessions and detect unusual connection attempts to compromised servers. Combined with TLS inspection, reputation filtering, and DNS analysis, this functionality significantly increases the chances of timely detection and blocking of malicious traffic.

It is important to note that firewalls cannot detect compromise within encrypted sessions, as such traffic appears to be legitimate. When HTTPS is used, an attacker must rely on forged certificates—forcing techniques such as SSL stripping or certificate spoofing. Due to these limitations, firewalls are often integrated with advanced threat detection and prevention systems (IDS/IPS), which will be discussed in the following section.

4.2.2 Firewall with IDS/IPS Gives Enhanced Protection

A firewall operates primarily as a statistical filter—focused on traffic rules and port-level control, but without visibility into packet content. However, when combined with intrusion detection and prevention systems (IDS/IPS), it becomes possible to identify anomalies within network traffic, including:

- Irregularities in the TLS handshake
- Unusual or suspicious DNS queries
- Attempts at ARP spoofing

This layered approach significantly improves the system's ability to detect and respond to Man-in-the-Middle (MITM) threats. Table 7 illustrates the relationship between firewall functionality and MITM mitigation strategies.

Table 7. Firewall and MITM Protection Overview

Element	Role in MITM Defense
Firewall	Blocks unauthorized connections, but cannot inspect encrypted MITM traffic
IDS/IPS	Analyzes traffic and detects anomalies
TLS/SSL	Prevents MITM if certificates are properly validated
User	If certificate warnings are ignored, MITM attacks may succeed

Source: Author

By combining firewalls with IDS/IPS systems, organizations achieve layered traffic analysis and proactive protection against MITM attacks. This significantly reduces the risk of compromise—even within encrypted communication channels.

5 MITM SCENARIOS

5.1 Recent Examples of MITM Attacks

The study by Čekerevac et al. (2025) presents a comprehensive overview of MITM attacks that occurred between 2007 and 2023. These cases will not be analyzed in detail here. Instead, the focus is placed on attacks that emerged in late 2023 and beyond. It is important to note that the exact percentage of MITM incidents varies depending on the source.

Arad (2024) reports that MITM attacks accounted for 23% of identity-related cyber incidents in 2024. According to the *Microsoft Digital Defense Report* (2024, p. 39), the intensity of password-based attacks reached a rate of 7,000 attempts per second.

5.1.1 Salt Typhoon Attack on the U.S. Telecommunications Companies

In 2024, the hacker group Salt Typhoon, reportedly linked to China (Krouse, McMillan, & Volz, 2024), conducted a sophisticated MITM

attack targeting major U.S. telecommunications providers, including AT&T, Verizon, Lumen Technologies, and T-Mobile (Kapko, 2025; Lyons, 2024; Israel & Young, 2025).

The attack was executed covertly, enabling unauthorized access to communication metadata—such as phone numbers, IP addresses, and timestamps (but not necessarily the content of calls or messages). This breach compromised the privacy of numerous users, including government officials and political campaign staff.

In response, the U.S. government established a dedicated incident response task force, while affected companies intensified collaboration with security agencies to strengthen infrastructure protection (Jaikaran, 2025).

5.1.2 Cozy Bear Attack on TeamViewer SE

In June 2024, German company TeamViewer SE—known for its remote monitoring and management (RMM) software used by managed service providers (MSPs) and IT departments to control servers, workstations, network devices, and endpoints—reported a breach of its corporate IT network by the Russian hacker group Cozy Bear (also known as APT29).

Access was gained through legitimate user credentials, although the method by which attackers obtained them was not disclosed. TeamViewer emphasized that its corporate IT systems are strictly segregated from the production environment of its remote access software, thereby preventing the attack from spreading to customer data. (Jones, 2024; Langley, 2024)

While the remote access product itself was not compromised, attackers accessed sensitive internal communications. The incident highlighted vulnerabilities in corporate IT infrastructure and the risks posed by advanced persistent threats (APTs) such as Cozy Bear. (Lakshmanan, 2024; Poireault, 2023)

5.1.3 Terrapin Attack on the SSH Protocol

In December 2023, researchers from Ruhr University Bochum discovered a vulnerability in the SSH protocol known as the *Terrapin Attack* (CVE-2023-48795). This flaw enables an MITM attacker to manipulate the initial messages of an SSH session using a technique called prefix truncation, which interferes with the negotiation of security extensions and silently downgrades session protection. (Palo Alto, 2024)

The attack specifically targets algorithms such as ChaCha20-Poly1305 and CBC with Encrypt-then-MAC, allowing the attacker to disable protections against keystroke timing attacks. According to unofficial estimates, nearly 11 million publicly accessible SSH servers were exposed to this risk (Toulas, 2024; Popovici, 2024).

Mitigation requires simultaneous updates on both the client and server sides, as unilateral patching is insufficient. Developers introduced a Strict Key Exchange option, which resets sequence counters and prevents packet injection during the unencrypted portion of the handshake. Additionally, a scanning tool for vulnerable hosts was released and made available via GitHub. (Mizrahi & Zohar, 2023; Ojha, 2023)

5.1.4 Iranian Hackers and the U.S. Presidential Campaign

In August 2024, Iranian hackers affiliated with the intelligence unit of the Islamic Revolutionary Guard Corps (IRGC) launched a spear-phishing attack targeting the presidential campaign of former President Donald Trump. The attack resulted in the compromise of a senior campaign official's account, enabling MITM interception of communications and the theft of sensitive documents. These documents were later leaked to the media via an anonymous account, including a research dossier on vice-presidential candidate JD Vance (Sharma, 2024).

According to a report by Microsoft, the Iranian actors used spoofed forwarded messages containing links that redirected traffic through attacker-controlled domains, thereby gaining access to confidential data (Sharma, 2024). U.S. agencies, including the FBI, CISA, and ODNI, confirmed Iran's involvement in attempts to compromise both major presidential campaigns—Trump's and the Biden-Harris team. (Kochi, 2024)

The incident raised concerns over foreign interference in the U.S. electoral process. In response, the campaign strengthened its cybersecurity protocols and initiated cooperation with federal authorities to investigate the breach. The attack underscored the urgent need for

enhanced digital protection in political campaigns (Aijaz, 2025).

5.1.5 OpenSSH Client Vulnerability CVE-2025-26465

In February 2025, researchers from the Qualys Threat Research Unit (2025) discovered a vulnerability in the OpenSSH client—CVE-2025-26465—which enables an MITM attack when the `VerifyHostKeyDNS` option is enabled. Due to a coding flaw, certain return codes from the host key verification function were improperly handled, allowing an attacker to impersonate a legitimate server and compromise the integrity of the SSH session.

The vulnerability affects versions 6.8p1 through 9.9p1, and was particularly active on systems such as FreeBSD, where the VerifyHostKeyDNS option was enabled by default from September 2013 to March 2023. The attacker could exploit this flaw by exhausting the client's memory and manipulating DNS responses, causing the client to incorrectly trust a malicious host key.

OpenSSH-a developers released a patch—version 9.9p2—on the same day, addressing the issue. The fix was confirmed in the NVD database entry for CVE-2025-26465 (NIST, CVE-2025-26465 Detail, 2025). However, the incident raised broader concerns about security defaults in widely used SSH clients (Abbasi, 2025).

5.1.6 MITM Attacks via Malicious Wi-Fi Networks and DNS Spoofing

Throughout 2025, there has been a notable increase in MITM attacks within public network environments, particularly through evil twin Wi-Fi networks and DNS spoofing. Attackers deployed rogue access points that mimicked legitimate networks in cafés, hotels, and airports, intercepting user communications and harvesting sensitive data such as passwords and credit card numbers. These attacks proved especially effective due to weak encryption protocols and the automatic reconnection of devices to known SSIDs[14] (JumpCloud, 2025).

Security experts emphasized the need for user education and broader adoption of VPN solutions in public networks to mitigate these risks. DNS spoofing enables attackers to redirect traffic by injecting false DNS records, often leading users to malicious websites that appear legitimate (Rawat, 2025).

In 2024, a surge in incidents was observed where attackers positioned themselves functionally between the user and the target system—aiming to gain unauthorized access, monitor communications, modify content, or exfiltrate confidential data.

To illustrate the broader spectrum of MITM scenarios—which extend beyond simple traffic interception to include compromised software components, proxy phishing techniques, and session token abuse—the following section presents controlled simulations of such attacks.

5.2 Use of ICMP Protocol in MITM Attack Initialization

The Internet Control Message Protocol (ICMP) is a fundamental component of network communication, primarily designed for error signaling and diagnostics (Postel, 1981). Although not inherently malicious, ICMP can be weaponized during the preparatory phase of Man-in-the-Middle (MITM) attacks—especially when combined with techniques such as ARP spoofing and routing manipulation (MITRE ATT&CK, Adversary-in-the-Middle [T1557], n.d.-a).

In MITM scenarios, ICMP messages are used to trigger legitimate network reactions that allow the attacker to position themselves between two communicating parties. For example, by sending an ICMP Echo Request to a target device, the attacker may provoke an ARP request, creating an opportunity to inject a forged ARP reply and redirect traffic. This technique enables the attacker to silently intercept, modify, or forward packets without raising suspicion among end users (SANS Institute, 2020).

Additionally, ICMP Redirect messages—if not blocked at the firewall level—can be exploited to manipulate routing paths, redirecting traffic to compromised nodes. While modern operating systems and network devices often ignore ICMP Redirects by default, their presence in poorly

[14] The SSID (Service Set Identifier) is a sequence of up to 32 alphanumeric characters that uniquely identifies a specific Wi-Fi network (Nicole, 2025). It functions as the network's name, allowing devices to distinguish between multiple wireless access points within range.

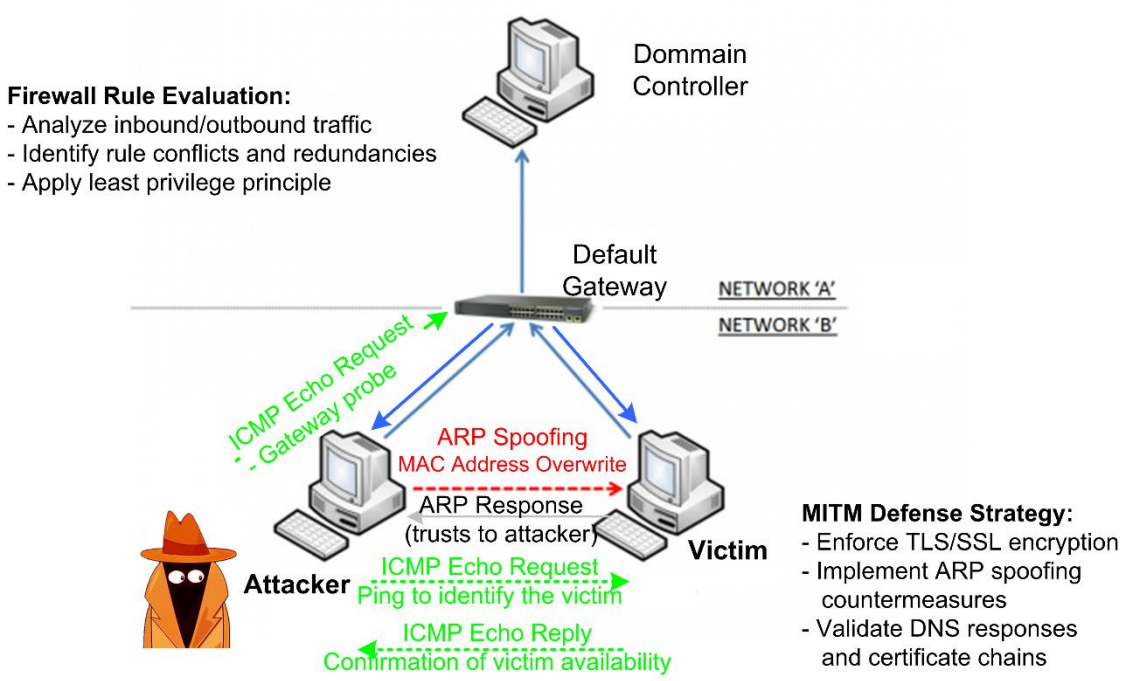
secured or misconfigured networks poses a significant security risk (MITRE ATT&CK, n.d.-b).

The misuse of ICMP in MITM preparation highlights the need for precise firewall rules that govern not only TCP and UDP traffic, but also ICMP messages—particularly those capable of triggering unintended network behavior. In this context, it is recommended to explicitly filter ICMP Redirect messages and monitor anomalous ICMP patterns that may indicate an attempt to establish a MITM position.

The diagram in Figure 3 illustrates the following stages:

- *Phase 1 – ICMP Echo* Request/Reply: Represented by arrows between the attacker and the victim, this phase initiates network activity.
- *Phase 2 – ARP Exchange:* ICMP communication triggers an ARP request from the victim, creating an opportunity for manipulation.
- *Phase 3 – Spoofed ARP Reply:* The attacker sends a forged ARP response, positioning themselves between the victim and the gateway.
- *Phase 4 – MITM Positioning:* Bidirectional arrows indicate interception and forwarding of packets.
- *Phase 5 – ICMP Redirect:* Shown as an additional arrow toward a compromised node, this phase reflects route manipulation. By sending an ICMP Redirect message, the attacker can permanently reroute traffic from the compromised host, establishing a stable MITM position without relying further on ARP spoofing.

These stages collectively illustrate how ICMP and ARP protocols can be sequentially exploited to establish a MITM position. The diagram highlights the attacker's strategic use of legitimate network mechanisms to silently intercept and manipulate traffic, underscoring the critical importance of protocol-level visibility and precision in defensive configurations.



MITM Initialization Diagram Based on ICMP and ARP Spoofing
Source: Author based on (CoreLabs Team, 2020)

This visualization clarifies the technical logic of the attack and highlights key intervention points—particularly through firewall rules, ARP monitoring, and ICMP filtering.

5.3 Simulation of a DNS Spoofing Attack on Viber

MITM attacks via DNS spoofing pose a serious threat to applications that rely on unencrypted DNS queries and insufficient TLS certificate

validation. The following scenario illustrates how an attacker can compromise communication between a user and Viber servers under real-world conditions.

5.3.1 Environment

Consider a user accessing Viber on a laptop or smartphone while sited in a café and connected to public Wi-Fi. The network appears legitimate but is deployed by an attacker using a tool such as *Wi-Fi Pineapple* to create a rogue access point. With utilities like *Ettercap* or *dnsspoof*, the attacker redirects DNS queries to a malicious DNS server that returns forged responses.

5.3.2 Attack Flow

1. The attacker sets up a rogue Wi-Fi access point using a familiar SSID (e.g., "Café_Free_WiFi"), creating the illusion of legitimacy.
2. The user's device automatically connects, unaware that the network is under attacker control.
3. All DNS requests sent by the user (e.g., for api.viber.com) are intercepted and replaced with spoofed IP addresses pointing to the attacker's server.
4. The Viber application attempts to establish a TLS connection with the server, but unknowingly communicates with the attacker.
5. The attacker presents a forged TLS certificate mimicking the legitimate Viber server.
6. If the application fails to properly validate the certificate, the connection is established, and the MITM attack succeeds.
7. If Viber detects an invalid certificate, the connection is terminated, and the user receives a warning, which may be ignored.

5.3.3 The Role of the Firewall in This Scenario

A firewall can play a significant role in mitigating this type of attack, but its effectiveness depends on the level of configuration and the presence of additional security layers.

A firewall can help if:

- It implements DNS filtering (e.g., via Pi-hole or DNS-over-HTTPS[15]), preventing interception and manipulation of DNS responses.
- It blocks unknown IP addresses and unauthenticated TLS handshakes[16] using advanced inspection mechanisms.
- It operates in conjunction with IDS/IPS systems that detect anomalies in DNS traffic and attempt to spoof certificates.

Firewall cannot help if:

- It allows unrestricted outbound traffic, enabling connections to malicious destinations.
- DNS queries are sent unencrypted (standard UDP port 53), making them easy to intercept.
- It lacks TLS inspection—in which case the firewall only sees that the connection is "allowed," without visibility into certificate content.

Table 8 summarizes the firewall's protective capabilities against MITM attacks in the context of a DNS spoofing scenario targeting the Viber application.

Table 8. Protection Against MITM Attacks Targeting the Viber Application

Element	Protects Against MITM?	Note
Firewall	Partially	Requires DNS filtering and TLS inspection
DNS over HTTPS	Yes	Prevents interception of DNS queries
TLS Validation	Critically important	If the application ignores certificate errors, MITM attack succeeds
User	Not effective if warnings are ignored	Human factor is often the weakest link

Source: Author

[15] The DNS over HTTPS protocol (commonly abbreviated as DoH) is formally defined in RFC 8484 (Hoffman & McManus, 2018)

[16] During the TLS handshake, cryptographic parameters are negotiated, certificates are exchanged, and a secure channel is established between the client and server.

TLS validation represents the critical point of protection—without it, all other layers (firewall, DNS filtering, IDS/IPS) can be bypassed. If the application fails to verify the certificate, or if the user ignores a warning about an invalid certificate, the attacker can successfully impersonate the server and gain full control over the communication.

5.4 Hypothetical MITM Scenario Involving WhatsApp

1. The user connects to a public Wi-Fi network.
2. The attacker deploys a rogue access point using a Wi-Fi Pineapple device.
3. WhatsApp attempts to establish a connection with its server.
4. The attacker intercepts the traffic and attempts to inject a forged TLS certificate.
5. A standard Layer 3/Layer 4 firewall can read the Server Name Indication (SNI) during the TLS handshake, but without TLS inspection, it cannot verify the certificate chain—allowing the forged certificate to pass undetected. Many applications, especially end-to-end encrypted services like WhatsApp, implement certificate pinning, which prevents TLS interception and limits firewall-level inspection capabilities.
6. WhatsApp detects the invalid certificate and terminates the connection.

Although the attack scenarios for Viber and WhatsApp are similar in intent, there are significant differences in their defensive architecture. Key distinctions are summarized in Table 9.

Table 9. Comparison of MITM Resistance in Viber and WhatsApp Applications

Aspect	Viber	WhatsApp
TLS Version	TLS 1.2/1.3 with inconsistent validation	TLS 1.3 with PSK for session resumption
Certificate Validation	Allows error bypassing in some versions	Strictly terminates connection on untrusted certificates
SNI Validation	SNI - not validated in certain versions	SNI checked before encryption
Resistance to Fake Certificates	Lower — accepts generic certificates	High — verifies CN/SAN and issuing authority
Updates and Patching	No publicly documented MITM-related CVEs	Rapid vulnerability fixes, e.g., CVE-2021-24027 (NIST, CVE-2021-24027 Detail, 2024)

Note: This table is based on publicly available statements and reports as of August 2025. For the most recent information, consult official security advisories.

Source: Author

5.5 Simulation of a Potential MITM Attack on Port 443

If a firewall allows unrestricted inbound traffic on port 443 without additional inspection, an MITM attacker may exploit this configuration using TLS stripping[17] to intercept and manipulate communication. In this scenario:

- The user attempts to access an HTTPS website (e.g., *https://example.com*)
- The attacker intercepts the request and responds with an HTTP version of the site (*http://example.com*)
- The user unknowingly communicates over an unencrypted HTTP connection
- – The attacker maintains a separate HTTPS session with the legitimate server and relays data, acting as a transparent intermediary

A properly configured firewall would:

- Detect unauthenticated certificates and downgrade attempts
- Block unencrypted requests directed at encrypted destinations (e.g., via HSTS policy enforcement[18])

[17] TLS stripping is an advanced MITM technique that prevents the automatic upgrade from HTTP to HTTPS, thereby keeping the user in an unencrypted communication mode.

[18] HTTP Strict Transport Security (HSTS) is a web server security policy that protects users from downgrade attacks and traffic interception (MITM). It enforces the use of HTTPS by instructing browsers to

- Alert administrators to TLS session manipulation attempts.

The diagram in Figure 4 illustrates the flow of a TLS stripping attack within an MITM scenario on port 443, highlighting the attacker's ability to downgrade secure connections and intercept sensitive data.

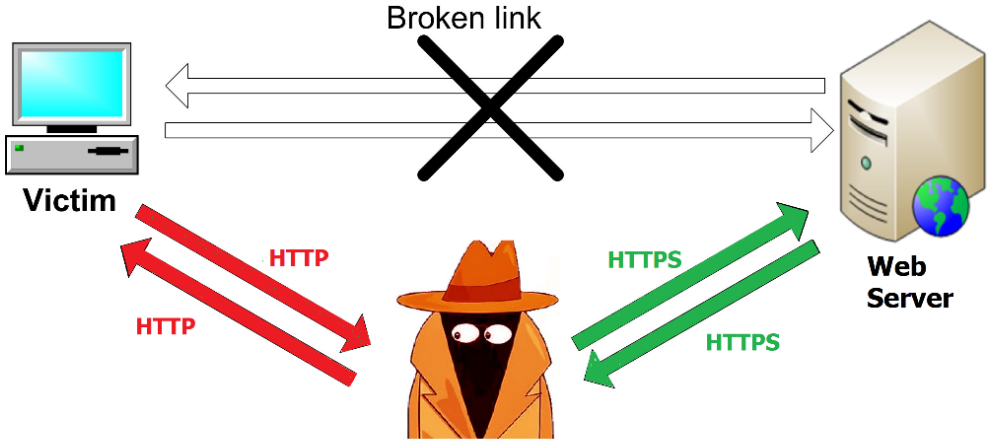


Figure 4. TLS Stripping Attack Flow in an MITM Scenario over Port 443

Source: Author

The attack unfolds as follows:

1. Victim attempts to access a website
   - The user types `example.com` - into the browser without specifying https://.
   - The browser sends an initial request over HTTP (port 80), as no HSTS policy is cached.
2. - Attacker intercepts the HTTP request
   - Positioned in an MITM role (e.g., via a rogue Wi-Fi access point), the attacker captures the request.
   - Instead of allowing the browser to upgrade to HTTPS, the attacker keeps the communication on HTTP.
3. - Attacker establishes an HTTPS session with the server
   - The attacker connects to `example.com` via HTTPS (port 443).
   - The server responds with encrypted content, such as a login form.
4. Attacker decrypts and converts the response to HTTP
   - The attacker presents the user with an unencrypted version of the site — visually identical, but insecure.
   - The user sees the form and enters credentials, unaware that the page is served over HTTP.
5. - Victim submits data via HTTP-a
   - Login credentials are transmitted in plaintext to the attacker.
   - The attacker forwards them to the server over HTTPS — the server remains unaware that the user was deceived.

The victim made two critical errors: connecting to a malicious hotspot and failing to enter the full https:// prefix. Had the initial request been explicitly HTTPS, the browser would have established a secure session directly with the server. The attacker would not have been able to view or modify the content, as the traffic would have been encrypted from the outset. TLS stripping would have been ineffective unless the attacker possessed a valid certificate, which is extremely difficult without compromising a certificate authority (CA).

Recommended countermeasures are:

- Always enter the full URL with https://
- Use a VPN when connected to public networks
- Prefer websites that implement HSTS policies and maintain valid TLS certificates
- Enable HSTS and browser-level HTTPS-Only Mode.

communicate exclusively over encrypted channels when interacting with a specified domain.

6 CONCLUSION: FIREWALL AND PROTECTION AGAINST MITM ATTACKS

Based on the analysis conducted, it can be concluded that a firewall can be effectively utilized in defending against MITM attacks—providing a positive answer to the first research question (RQ1).

Furthermore, the analysis addresses the second research question (RQ2): In MITM scenarios such as DNS spoofing, firewall protection can play a significant role in safeguarding users—provided it is properly configured and integrated into a broader security strategy.

This leads to the following conclusions:

- *The firewall is not self-sufficient*: Basic access rules and port filtering alone are insufficient to prevent sophisticated MITM attacks.
- *DNS filtering and DoH*: These mechanisms prevent interception and manipulation of DNS queries, eliminating one of the most common MITM vectors.
- *TLS validation is critical*: Applications must rigorously verify server certificates—without this, an attack may succeed even with firewall protection in place.
- *Advanced firewall features* (e.g., IDS/IPS, TLS inspection, network segmentation) enable anomaly detection and the blocking of suspicious traffic.
- *The human factor remains a persistent vulnerability*: Ignoring certificate warnings or connecting to unknown networks can undermine technical safeguards.
- The firewall can be an effective tool against MITM attacks, but only as part of a multi-layered security architecture that includes:
  - Encrypted DNS queries
  - TLS certificate validation
  - Anomaly detection
  - User education

In the context of applications employing end-to-end encryption—such as Viber—the firewall can help prevent access to rogue servers, but it cannot replace the internal security mechanisms of the application itself.

WORKS CITED

AAG. (2025, July 1). *The latest 2025 cyber crime statistics (updated July 2025).* https://aag-it.com/the-latest-cyber-crime-statistics/

Abbasi, S. (2025, February 18). *Qualys TRU discovers two vulnerabilities in OpenSSH: CVE-2025-26465 & CVE-2025-26466.* https://blog.qualys.com/vulnerabilities-threat-research/2025/02/18/qualys-tru-discovers-two-vulnerabilities-in-openssh-cve-2025-26465-cve-2025-26466

Aijaz, D. (2025, January 1). *Year-end analysis: Man-in-the-middle attacks in the US in 2024.* https://www.purewl.com/man-in-the-middle-attacks-in-the-us-in-2024/

APWG. (2025, July 2). *Phishing activity trends reports: 1st quarter 2025.* https://docs.apwg.org/reports/apwg_trends_report_q1_2025.pdf

Arad, R. (2024, November 20). *6 ways to prevent man-in-the-middle (MITM) attacks.* https://www.memcyco.com/6-ways-to-prevent-man-in-the-middle-mitm-attacks/

Astra Security. (2023, July 7). *13 man-in-the-middle attack statistics you must know about.* https://securityescape.com/man-in-the-middle-attack-statistics/

Cekerevac, Z., Cekerevac, P., Prigoda, L., & Naima, F. A. (2025, January 15). Security risks from the modern man-in-the-middle attacks. *MEST Journal, 13*(1), 34–51. https://doi.org/10.12709/mest.13.13.01.04

Cekerevac, Z., Dvorak, Z., Prigoda, L., & Cekerevac, P. (2017a). Internet of things and the man-in-the-middle attacks – Security and economic risks. *MEST Journal, 5*(2), 15–25. https://doi.org/10.12709/mest.05.05.02.03

Cekerevac, Z., Dvorak, Z., Prigoda, L., & Cekerevac, P. (2017). Techno-economic aspect of the man-in-the-middle attacks. *Communications, 2*, 166–172. https://doi.org/10.26552/com.C.2017.2.166-172

Citakovic, S. (2023, May 23). *10 SQL injection attacks statistics to know in 2023.* https://securityescape.com/sql-injection-attacks-statistics/

CoreLabs Team. (2020, May 22). *MS15-011 – Microsoft Windows Group Policy real exploitation via a SMB MiTM attack.* https://www.coresecurity.com/core-labs/articles/ms15-011-microsoft-windows-group-policy-real-exploitation-via-a-smb-mitm-attack

CVEdetails. (2025). *Security vulnerabilities, CVEs published in 2024.* https://www.cvedetails.com/vulnerability-list/year-2024/vulnerabilities.html

Dingledine, R., Mathewson, N., & Syverson, P. (2004). Tor: The second-generation onion router. In *Proceedings of the 13th USENIX Security Symposium* (Vol. 13, p. 17). USENIX Association. https://doi.org/10.5555/1251375.1251396

Hackmanac. (2024, July 24). *Global cyber attacks report 2024.* https://hackmanac.com/hackmanac-global-cyber-attacks-report-2024

Hlapisi, N. (2023, July 16). *Vulnerabilities and attacks on Bluetooth LE devices—Reviewing recent info.* https://www.allaboutcircuits.com/technical-articles/vulnerabilities-and-attacks-on-bluetooth-le-devicesreviewing-recent-info/

Hoffman, P., & McManus, P. (2018, October). *DNS Queries over HTTPS (DoH).* https://www.rfc-editor.org/rfc/rfc8484.html

IBM. (2025). *Cost of a data breach.* https://www.ibm.com/downloads/documents/us-en/131cf87b20b31c91

Israel, K., & Young, R. (2025, January 10). *Verizon provides update on Salt Typhoon matter.* https://www.verizon.com/about/news/verizon-provides-update-salt-typhoon-matter

Jackson, M. (2024, November 8). *The state of SQL injection.* https://www.aikido.dev/blog/the-state-of-sql-injections

Jaikaran, C. (2025, January 23). *Salt Typhoon hacks of telecommunications companies and federal response implications.* https://www.congress.gov/crs-product/IF12798

Jones, L. (2024, July 01). *TeamViewer Network Breach Linked to Russian Hackers.* https://winbuzzer.com/2024/07/01/teamviewer-confirms-internal-it-system-compromise-xcxwbn/

JumpCloud. (2025, March 7). *What is an evil twin WiFi attack?* https://jumpcloud.com/it-index/what-is-an-evil-twin-wifi-attack

Kapko, M. (2025, January 7). *AT&T, Verizon say they evicted Salt Typhoon from their networks.* https://www.cybersecuritydive.com/news/att-verizon-salt-typhoon/736680/

Kaspersky. (2025, June 20). *What is a VPN? How it works, types, and benefits.* https://www.kaspersky.com/resource-center/definitions/what-is-a-vpn

Kochi, S. (2024, August 19). *Intelligence groups say Iran behind hacking attempts in Biden-Harris and Trump campaign.* https://eu.usatoday.com/story/news/politics/elections/2024/08/19/fbi-concludes-iran-hacking-attempt-trump/74866004007/

Krouse, S., McMillan, R., & Volz, D. (2024, September 26). *China-linked hackers breach U.S. internet providers in new 'Salt Typhoon' cyberattack. The Wall Street Journal.* https://www.wsj.com/politics/national-security/china-cyberattack-internet-providers-260bd835

Lakshmanan, R. (2024, June 28). *TeamViewer detects security breach in corporate IT environment.* https://thehackernews.com/2024/06/teamviewer-detects-security-breach-in.html

Langley, M. (2024, July 2). *TeamViewer confirms breach by notorious Russian hacking group Cozy Bear.* https://dailysecurityreview.com/security-spotlight/teamviewer-confirms-breach-by-notorious-russian-hacking-group-cozy-bear/

Lyons, J. (2024, December 30). *More telcos confirm China Salt Typhoon security breaches as White House weighs in.* https://www.theregister.com/2024/12/30/att_verizon_confirm_salt_typhoon_breach/

McAfee. (2016). *McAfee Labs threats report.* Intel Security.

Microsoft. (2024). *Microsoft digital defense report 2024.* https://cdn-dynmedia-1.microsoft.com/is/content/microsoftcorp/microsoft/final/en-us/microsoft-brand/documents/Microsoft%20Digital%20Defense%20Report%202024%20%281%29.pdf

MITRE ATT&CK. (n.d.-a). *Adversary-in-the-middle (T1557).* https://attack.mitre.org/techniques/T1557/

MITRE ATT&CK. (n.d.-b). *Non-application layer protocol (T1095).* https://attack.mitre.org/techniques/T1095/

Mizrahi, Y., & Zohar, M. (2023, December 25). *SSH protocol flaw – Terrapin attack CVE-2023-48795: All you need to know.* https://jfrog.com/blog/ssh-protocol-flaw-terrapin-attack-cve-2023-48795-all-you-need-to-know/

Morgan, S. (2023, October 3). *Software supply chain attacks to cost the world $60 billion by 2025.* https://cybersecurityventures.com/software-supply-chain-attacks-to-cost-the-world-60-billion-by-2025/

Morgan, S. (2025, March 12). *Global ransomware damage costs predicted to hit.* https://elastio.com/wp-content/uploads/2025/04/RANSOMWARE-REPORT-2025-final.pdf

Mullvad. (2025, March 27). *Successful security assessment of our Android app.* https://mullvad.net/en/blog/successful-security-assessment-of-our-android-app/

NETSCOUT. (2024). *DDoS threat intelligence report – 1H 2024 (Issue 13).* https://www.netscout.com/threatreport/wp-content/uploads/2024/09/TR_1H2024_Web.pdf

Nickfetrat, F. (2024, October 9). *iptables vs nftables: What's new in Linux firewalling?* https://dev.to/farshad_nick/iptables-vs-nftables-whats-new-in-linux-firewalling-4a36

Nicole, S. (2025, July 4). *What is an SSID & why naming conventions matter.* https://exactlyhowlong.com/what-is-an-ssid-why-naming-conventions-matter/

NIST. (2024, November 21). *CVE-2021-24027 detail.* https://nvd.nist.gov/vuln/detail/CVE-2021-24027

NIST. (2025, June 2). *CVE-2025-26465 detail.* https://nvd.nist.gov/vuln/detail/CVE-2025-26465

Ojha, D. (2023, December 22). *SSH prefix truncation vulnerability used in Terrapin attacks (CVE-2023-48795).* https://threatprotect.qualys.com/2023/12/22/ssh-vulnerability-used-in-terrapin-attacks-cve-2023-48795/

Okoruwa, S., & Chapman, S. (2025, April 25). *25 ransomware statistics, facts & trends in 2025.* https://www.cloudwards.net/ransomware-statistics/

OWASP. (2025). *XML security cheat sheet.* https://cheatsheetseries.owasp.org/cheatsheets/XML_Security_Cheat_Sheet.html#man-in-the-middle-mitm-attack

Palatty, N. J. (2025, June 20). *How many cyber attacks per day: The latest stats and impacts in 2025.* https://www.getastra.com/blog/security-audit/how-many-cyber-attacks-per-day/

PAN PSIRT. (2024, January 09). *CVE-2023-48795 Impact of Terrapin SSH Attack.* https://securityadvisories.paloaltonetworks.com/CVE-2023-48795

Poireault, K. (2023, December 14). *Cozy Bear hackers target JetBrains TeamCity servers in global campaign. Infosecurity Magazine.* https://www.infosecurity-magazine.com/news/cozy-bear-russia-jetbrains-teamcity/

Postel, J. (1981, September). *RFC 792: Internet control message protocol.* https://datatracker.ietf.org/doc/html/rfc792

Qualys Threat Research Unit. (2025, February 19). *2023 Qualys TruRisk research report.* https://www.qualys.com/forms/tru-research-report/

Rawat, P. (2025, August 12). *DNS spoofing vs. MITM attack*. InfoSecTrain. https://www.infosectrain.com/blog/dns-spoofing-vs-mitm-attack/

Red Hat. (2024, October 6). *CVE-2023-48795.* https://access.redhat.com/security/cve/cve-2023-48795

SANS Institute. (2020). *ICMP abuse in network attacks.* https://www.sans.org

Sharma, S. (2024, August 12). *Trump campaign suffers sensitive data breach in alleged Iranian hack.* https://www.csoonline.com/article/3485643/trump-campaign-suffers-sensitive-data-breach-in-alleged-iranian-hack.html

Smith, G. (2025, June 4). *Top +35 DDoS statistics (2025).* https://www.stationx.net/ddos-statistics/

Snape, G. (2025, February 19). *Supply chain cyber attacks surge over 400%, expected to continue rising – Cowbell report.* https://www.insurancebusinessmag.com/us/news/cyber/supply-chain-cyber-attacks-surge-over-400-expected-to-continue-rising--cowbell-report-525369.aspx

SOCRadar. (2024, June 13). *Phishing in 2024: 4,151% increase since launch of ChatGPT; AI mitigation methods.* https://socradar.io/phishing-in-2024-4151-increase-since-chatgpt/

Spring, T. (2016, August 11). *Bluetooth hack leaves many smart locks, IoT devices vulnerable.* https://threatpost.com/bluetooth-hack-leaves-many-smart-locks-iot-devices-vulnerable/119825/

SSL Insights. (2025, May 30). *Phishing statistics for 2025: Latest figures and trends.* https://sslinsights.com/phishing-statistics/

StormWall. (2025, May 28). *What's new in the world of DDoS? StormWall's Q1 2025 report.* https://stormwall.network/resources/blog/ddos-report-q1-2025

Threat Hunter Team. (2025, February 20). *Ransomware 2025: Attacks keep rising as threat shows its resilience.* https://www.security.com/threat-intelligence/ransomware-trends-2025

Toulas, B. (2024, January 3). *Nearly 11 million SSH servers vulnerable to new Terrapin attacks.* https://www.bleepingcomputer.com/news/security/nearly-11-million-ssh-servers-vulnerable-to-new-terrapin-attacks/

Vahab, A. B. (2025, May 06). *OWASP IoT Top 10 Vulnerabilities (2025 Updated)*. Wattlecorp Cybersecurity Labs: https://www.wattlecorp.com/owasp-iot-top-10/

Verizon. (2025). *2025 data breach investigations report.* https://www.verizon.com/business/resources/reports/dbir/

Wabuge, D. (2023, July 7). *13 man-in-the-middle attack statistics you must know about.* https://securityescape.com/man-in-the-middle-attack-statistics/

Watlecorp. (2025, May 6). *OWASP IoT top 10 vulnerabilities (2025 updated).* https://www.wattlecorp.com/owasp-iot-top-10/