Diving Into Zero-Touch Network Security: Use-Case Driven Analysis

Li Yang ¹, Mirna El Rajab ², Abdallah Shami ², and Sami Muhaidat ²

¹Western University
²Affiliation not available

October 31, 2023

Abstract

The sixth generation of wireless networks (6G) will require network automation to meet the rapidly increasing demands for high data rate services, ultra-low latency, massive connectivity, and seamless integration with emerging technologies, while effectively reducing operating costs. To address these demands, the concept of Zero-Touch Networks (ZTNs) has been proposed, where Artificial Intelligence (AI) and Machine Learning (ML) play crucial roles in optimizing network performance, enabling intelligent decision-making, and ensuring efficient resource allocation. However, the implementation of ZTNs is subject to security challenges that may hinder their development and deployment. In particular, two critical challenges arise: the need for human expertise in developing AI/ML-based security mechanisms, and the threat of specific attacks targeting AI/ML models. In this survey paper, a comprehensive review of the security vulnerabilities and issues with ZTNs is conducted. Additionally, potential automated solutions to ZTN security concerns, with a specific focus on leveraging Automated ML (AutoML) technologies, are investigated. Two case studies are conducted to address security issues in ZTNs and further corroborate our findings: the development of autonomous intrusion detection systems and the creation of defense mechanisms against Adversarial ML (AML) attacks. Finally, some of the challenges and future research directions for the development of ZTN security approaches are discussed.
Diving Into Zero-Touch Network Security: Use-Case Driven Analysis

Li Yang, Member, IEEE, Mirna El Rajab, Member, IEEE, Abdallah Shami, Senior Member, IEEE, and Sami Muhaidat, Senior Member, IEEE

Abstract—The sixth generation of wireless networks (6G) will require network automation to meet the rapidly increasing demands for high data rate services, ultra-low latency, massive connectivity, and seamless integration with emerging technologies, while effectively reducing operating costs. To address these demands, the concept of Zero-Touch Networks (ZTNs) has been proposed, where Artificial Intelligence (AI) and Machine Learning (ML) play crucial roles in optimizing network performance, enabling intelligent decision-making, and ensuring efficient resource allocation. However, the implementation of ZTNs is subject to security challenges that may hinder their development and deployment. In particular, two critical challenges arise: the need for human expertise in developing AI/ML-based security mechanisms, and the threat of specific attacks targeting AI/ML models. In this survey paper, a comprehensive review of the security vulnerabilities and issues with ZTNs is conducted. Additionally, potential automated solutions to ZTN security concerns, with a specific focus on leveraging Automated ML (AutoML) technologies, are investigated. Two case studies are conducted to address security issues in ZTNs and further corroborate our findings: the development of autonomous intrusion detection systems and the creation of defense mechanisms against Adversarial ML (AML) attacks. Finally, some of the challenges and future research directions for the development of ZTN security approaches are discussed.


I. INTRODUCTION

The evolution of wireless communication technology began in 1980 with the launch of the first generation of cellular networks (1G). Ever since, telecommunications and networking have witnessed significant advancements, manifested by the emergence of the 2G, 3G, and 4G cellular network paradigms. The deployment phase of the fifth-generation (5G) wireless technology began in 2020, with its software-based evolution expected to be fully completed by 2025 [1]. 5G enables enhanced Mobile Broadband (eMBB) with peak data speeds of up to 10 Gbps. Additionally, ultra-Reliable Low Latency Communication (uRLLC) reduces latency to less than one millisecond, while massive Machine Type Communication (mMTC) allows for over 100 times more devices per unit area than 4G [2].

The key enablers of 5G networks involve Software-Defined Networking (SDN), Network Function Virtualization (NFV), Network Slicing (NS), and Multi-access Edge Computing (MEC) [3]. With SDN, networks can be dynamically configured and monitored from a central location, enabling holistic and global network management regardless of the underlying technology. NFV decouples network functions from hardware, allowing for rapid deployment of new services, flexible scalability to meet customer demands, and dynamic network configuration and monitoring. NS enables the creation and deployment of multiple logical networks on the same physical infrastructure, while MEC facilitates the relocation of computing resources from the cloud servers to the edge devices and nodes [3]. This abstraction of physical resources into virtual and logical settings enables on-demand automatic learning and management capabilities.

Researchers have already begun envisioning the sixth generation (6G) of mobile wireless networks, despite the fact that 5G deployment/coverage is not yet complete. It is expected that 6G standardization will commence in 2026 [2]. In spite of the development of various solutions and enablers such as SDN, NFV, MEC, and NS to enhance network performance, achieving fully autonomous network and service management remains challenging, as human intervention is still necessary for current networks.

Recently, several concepts or architectures have been proposed for network automation and optimization, such as Self-Organizing Network Management (SON), Intent-Based Network Management (IBN), and Autonomic Network Management (ANM) [4]. More recently, Zero-touch Network and Service Management (ZSM) were proposed by the European Telecommunications Standards Institute (ETSI) [5]. ZSM, or Zero-Touch Network (ZTN), is defined as a fully autonomous network management architecture that minimizes the need for human intervention. On the other hand, in the ZSM paradigm, a network can evaluate its present state, interpret it, and provide recommendations for potential reconfigurations [4].

It should be pointed out that the network configurations and policies can be deployed and managed automatically in ZTNs. The networking and communication scientific community anticipates that Artificial Intelligence (AI) and Machine Learning (ML) techniques will play a vital role in fully automating the management and orchestration of 6G networks [2].

AI/ML techniques have become key enablers for ZTNs/6G networks, realizing the rapid and accurate processing of massive volumes of network data to achieve network automation and optimization [6]. The implementation of ML approaches

Li Yang, Mirna El Rajab, and Abdallah Shami are with the Department of Electrical and Computer Engineering, Western University, London, ON N6A 3K7, Canada (e-mails: lyang339@uwo.ca, meralrajab@uwo.ca, abdallah.shami@uwo.ca)

Sami Muhaidat is with the Department of Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi 127788, UAE (e-mail: sami.muhaidat@ku.ac.ae)
In network and service management leads to significant improvements in service efficiency, performance, and time management [3]. AI/ML methods can provide or enhance a wide range of network services and management functionalities, such as network behavior analysis, anomaly detection, traffic classification and forecasting, mobility prediction, and resource allocation.

In spite of the widespread adoption of AI/ML techniques in 5G/6G networks, developing and deploying ML algorithms still requires extensive domain expertise and human effort [7]. To realize autonomous networks and ZTNs, Automated Machine Learning (AutoML) techniques provide potential solutions by enabling automated network data analytics for data-driven network services [8]. AutoML enables the automation of various data analytics and ML procedures, including automated data pre-processing, automated feature engineering, automated model selection, Hyper-Parameter Optimization (HPO), and automated model updating [9]. Automated data preprocessing and feature engineering aim to enhance network data quality for improved analytics results, whereas automated model selection and HPO produce optimized ML models with optimal performance. Moreover, real-world networks typically operate in ever-changing environments, leading to model drift or data distribution changes. Automated model updating is a potential solution to address model drift issues and maintain the performance of AI/ML models [10]. Overall, AutoML techniques are effective solutions for realizing ZTNs.

Critical self-management functionalities in ZTNs include self-configuration, self-healing, self-optimization, and self-protection [11]. Among these functionalities, both self-healing and self-protection are particularly relevant to network security concerns, making network security mechanisms critical components in ZTNs. ZTN security exemplifies the concept that security should be an integral component of the network infrastructure [12]. By incorporating security measures into the network itself, management and maintenance are significantly simplified, since no additional hardware or software is required.

Cybersecurity issues are primary challenges in traditional networks and ZTNs. Traditional networks face various cyber threats, such as Denial of Service (DoS), eavesdropping, Man-in-the-Middle (MITM), web attacks, and malware. Apart from traditional network security threats, the adoption of ZTN technologies in Beyond 5G (B5G)/6G networks is envisioned to introduce additional security challenges and increased attack surfaces. These include threats to open Application Programming Interface (API), intent, closed-loop network automation, programmable network technology, and AI/ML models [3] [11]. Among these security threats, AI/ML-based attacks, or Adversarial Machine Learning (AML) attacks, are expected to introduce critical challenges in ZTNs/6G networks. AML attacks exploit the vulnerabilities of AI/ML models, causing them to make incorrect predictions/decisions and leading to security breaches and unauthorized access to sensitive data [13]. Therefore, it is crucial to develop effective security mechanisms that can detect, mitigate, and prevent security threats in ZTNs [12].

Although there is an increasing popularity in ZTN research in the area of 6G networks, a comprehensive survey on ZTN security has yet to be conducted. This survey paper aims to bridge the gap between ZTN techniques and cybersecurity applications by reviewing existing challenges and potential solutions between these two domains. To the best of our knowledge, this survey represents the first comprehensive review of ZTN security and automated cybersecurity methods by introducing AutoML techniques. Specifically, this survey paper provides the following contributions:

i) A comprehensive review of security issues and cyber-attacks in 5G/6G networks, covering traditional cybersecurity challenges, ZTN security threats, and AI/ML model security issues.

ii) A review of potential solutions to ZTN security issues, including general cybersecurity procedures, AutoML techniques, and AML defense mechanisms.

iii) Two case studies that apply AutoML techniques to practical ZTN security tasks and conduct the cyber-defense exercise against AML attacks; the implementation code will be publicly available on GitHub.

iv) A discussion of the open challenges and research directions of ZTN security methods.

This manuscript is organized as follows: Section II provides an overview of the ZTN architecture and security requirements. Section III presents the vulnerabilities, cyber-attacks, and general security mechanisms for traditional networks. Section IV discusses the vulnerabilities and potential solutions specifically for ZTNs/6G networks. Section V introduces AI/ML & AutoML techniques, their security threats (i.e., AML attacks), and potential solutions. Section VI presents two case studies of the autonomous ZTN security framework using AutoML techniques and the AML cyber-defense exercise. Section VII presents the challenges and future directions of the ZTN security research domains. Section VIII concludes the paper.

II. ZTN Security Overview

A. ZTN Overview

ZTNs have emerged as a solution for fully automating network operations, providing self-configuration, self-monitoring, self-healing, and self-optimization [13]. The adoption of ZTNs is driven by the demand for virtualized network functionality and the preference for software-based solutions over hardware-based alternatives [3]. They are designed to operate, conduct commercial activities, self-sustain, recover from disruptions, and perform other network-related tasks without human intervention. The deployability of ZTNs is facilitated by various APIs and End-to-End (E2E) programmability [3]. ZTNs are anticipated to play a crucial role in future networks, enabling the efficient distribution of network resources to meet the specific requirements of various industries and individuals [4]. The automation capabilities provided by ZTNs are essential for analyzing and responding to customer-specific needs, ensuring the desired Quality of Experience (QoE). In addition, ZTNs

can evolve and upgrade through E2E automation, incorporating and updating new features that are required for practical applications.

ZTNs require a high level of automation, highlighting the importance of AI/ML techniques in transforming network management to achieve self-adaptation and self-reaction in highly dynamic environments. To facilitate network automation, Release 17 of the 3rd Generation Partnership Project (3GPP) includes a NFV/SDN approach via the Network Data Analytics Function (NWDAF) [15]. The ZSM paradigm established by ETSI represents an advancement in network orchestration via an E2E framework that leverages AI/ML and data analytics techniques [5]. By eliminating tightly coupled management systems, ZSM enables a flexible service management architecture and defines the architectural building blocks for implementing the complex management services required to realize ZTNs.

B. ZTN Security Architecture

In the field of network and service management, cybersecurity automation can be achieved through the use of AI/ML techniques in combination with closed-loop operations [12]. Each management domain within the ZSM architecture contains a data analytics service that enables closed-loop network security and optimization operations. An example of a fully automated network security framework for ZSM architectures is proposed in the "Intelligent security and pervasive trust for 5G and beyond" (INSPIRE-5Gplus) project, funded by the European Union (EU) [16].

Closed-loop operations are essential enablers of network automation within the ZSM architecture. Figure 1 illustrates a domain-level security closed-loop ZSM framework, which consists of five stages: observation, orientation, decision, action, and learning [12] [16]. In the first stage, data is observed and collected from various data sources by the security agents. The collected data is then oriented and analyzed by the security analytics service to comprehend its meaning and significance, which may involve data filtering and correlation analysis. In the subsequent stage, the decision engine uses AI/ML algorithms to recognize data patterns and make informed decisions or predictions for network optimization. Upon identifying the root causes and providing recommendations, the security orchestrator implements environment- and situation-appropriate actions or plans. Lastly, the system learns from the detected patterns and consequences to improve future reactions or decisions.

E2E management domains are responsible for managing service and resource requests across different domains [12]. A management domain refers to a collection of services and resources organized based on various constraints, such as operations, functions, and deployment. Communication within domains is facilitated by the domain integration fabric, while communication between domains and E2E management domains is enabled by the inter-domain integration fabric. Within the ZSM architecture illustrated in Fig. 1, a management domain consists of a security data collector for data observation and collection, a security analytics engine for orientation, a decision engine enabled by AI/ML methods for decision-making, a policy and Security Service Level Agreement (SSLA) manager for governance, a security orchestrator and trust manager for action-taking, and data services for knowledge learning.

C. ZTN Security Requirements

The ZTN framework reference architecture plays an important role in the networking and cybersecurity landscape. As the world transitions toward more automated and intelligent networks, the need for stringent security measures becomes increasingly crucial. The ZTN security frameworks aim to reduce the risk of human error and create an efficient and secure network environment. For ZTN security frameworks, several essential security requirements must be met [3] [11] [17]:

i) Data Protection and Integrity: The ZSM framework must support data protection mechanisms that cover data in use, in transit, and at rest. Ensuring the security of data management and data integrity is essential for ZSM frameworks. In addition, the ZSM framework must ensure the availability of network data, infrastructure resources, and management functions.

ii) Privacy Preservation: The ZSM framework should incorporate personal data privacy features that conform to the privacy-by-design and privacy-by-default principles. These principles ensure that privacy is built into the architecture from the outset, with the highest level of privacy settings applied by default and without user intervention.
Access Control and Security Policy Enforcement: The ZSM framework should enable authenticated service users to authorize service access and follow security policies. This capability allows for the automated implementation of appropriate security policies based on the status of individual management services in relation to security requirements. Streamlining access control enhances overall security posture.

Intrusion Detection, Prevention, and Mitigation: The ZSM framework must support automated detection, recognition, prevention, and mitigation of attacks and incidents. These capabilities are crucial for swiftly identifying and mitigating potential security threats, thereby minimizing the impact of cyber-attacks on the network and its services.

AI/ML Decision Supervision and Auditing: To safeguard against vulnerabilities and cyber-attacks, the ZSM platform reference architecture should enable the supervision and auditing of privacy and security decisions made by AI/ML techniques. This mechanism guarantees that AI/ML-driven decisions establish security requirements, mitigate potential risks, and recover compromised devices while remaining resilient to AI/ML-based attacks.

III. Traditional Network Security Threats and Countermeasures

A. Cyber-Threats in Traditional Networks

The rapid growth of the Internet and technological advancements have given rise to an alarming surge in cyber threats targeting various networks. As cyber-attacks become more sophisticated, it is crucial to understand the prevalent types of attacks in traditional networks. This section explores the evolution of cybersecurity in traditional networks and highlights various cyber threats that jeopardize their security.

Traditional networks, from 1G to 5G, have evolved considerably, with each generation introducing new capabilities and security features [18]. The 1G networks in the 1980s were analog cellular networks without Internet connectivity, making cybersecurity less of a concern. The 2G networks enabled data transmission and text messaging, while 3G networks provided faster data transmission rates [18]. Despite security improvements, threats such as DoS, MITM, and eavesdropping attacks emerged during the 2G and 3G eras and persist in modern networks [19].

The late 2000s witnessed the introduction of 4G networks, which brought new security features like Virtual Private Networks (VPNs) and advanced encryption methods [18]. However, 4G networks remained vulnerable to many cyber-attacks, including phishing, malware, and Distributed DoS (DDoS) attacks. The latest 5G networks offer high-speed data transmission rates and low latency, making them essential for applications like the Internet of Things (IoT) and autonomous vehicles [20]. Nevertheless, security remains a significant concern for 5G networks due to the increasing number of connected devices and the growing network complexity.

Common traditional cyber threats include DoS/DDoS, unauthorized access, eavesdropping, MITM, phishing, web attacks, and malware. DoS attacks aim to disrupt a network or service's availability by overwhelming it with excessive traffic, potentially causing significant losses [21]. DDoS attacks, a variant of DoS attacks, employ compromised devices to target a system, resulting in widespread network service disruption [22]. Unauthorized access involves accessing a system without permission or exceeding authorized access privileges, leading to data theft, manipulation, network disruption, and unauthorized information dissemination [23].

Eavesdropping attacks involve intercepting and monitoring sensitive information transmitted over a network [24], using techniques like wiretapping, sniffing, or rogue access points. These attacks can compromise sensitive data such as personal information, financial data, or intellectual property. MITM attacks occur when an attacker intercepts communication between two parties, potentially altering messages, impersonating one party, compromising sensitive data, or facilitating unauthorized system access [25].

Phishing attacks use deceptive communication, often via email, to trick recipients into revealing sensitive information or downloading malware [26]. Attackers often employ urgency or emotional appeals to manipulate victims. Web-based attacks targeting web applications in traditional networks include Structured Query Language (SQL) injection attacks and Cross-Site Scripting (XSS) attacks [21]. SQL injection attacks exploit vulnerabilities in web applications using SQL databases, granting unauthorized access to sensitive information or database modification. XSS attacks inject malicious code into web pages viewed by other users, enabling attackers to steal sensitive information or manipulate target actions.

Malware, short for malicious software, includes various harmful software types designed to infiltrate, damage, or gain unauthorized access to computer systems or networks [25]. Common malware forms include viruses, worms, Trojan horses, ransomware, and spyware. These malware forms can steal sensitive information, disrupt network operations, or cause significant financial losses by encrypting files and demanding ransom for release.

The description and impacts of the aforementioned cyber threats in traditional networks are summarized in Table I. To summarize, traditional networks face various cyber threats that can cause significant harm to organizations and individuals. As technology continues to advance and cyber-attacks become more sophisticated, it is essential to stay vigilant and implement robust cybersecurity measures to protect these networks. This includes continuously updating security protocols, implementing encryption, educating users on potential threats, and investing in advanced intrusion detection and prevention systems.

B. General Cybersecurity Procedures

To enhance defense against cyber-attacks, comprehensive cybersecurity systems incorporate four essential steps: anomaly detection, root cause analysis, system remediation, and intrusion prevention.

1) Attack/Intrusion Detection: Ensuring the reliability and availability of 5G and beyond networks requires the identification and prediction of abnormal behaviors resulting from
TABLE I
THE DESCRIPTION OF TRADITIONAL CYBER THREATS.

<table>
<thead>
<tr>
<th>Category</th>
<th>Attack Type</th>
<th>Description</th>
<th>Potential Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>DoS</td>
<td>Overwhelming network or service with excessive traffic.</td>
<td>System unavailability and losses for organizations.</td>
</tr>
<tr>
<td></td>
<td>DDoS</td>
<td>Using a network of compromised devices to launch attacks.</td>
<td>Widespread disruption and losses for organizations.</td>
</tr>
<tr>
<td>Unauthorized</td>
<td>Unauthorized</td>
<td>Accessing a system, network, or data without proper</td>
<td>Data theft, network disruption, and information</td>
</tr>
<tr>
<td>Access</td>
<td>Access</td>
<td>authorization.</td>
<td>dissemination.</td>
</tr>
<tr>
<td>Interception</td>
<td>Eavesdropping</td>
<td>Intercepting and listening to sensitive information</td>
<td>Information leakage and exploitation of intercepted</td>
</tr>
<tr>
<td></td>
<td>MitM</td>
<td>transmitted over a network.</td>
<td>data.</td>
</tr>
<tr>
<td>Deception</td>
<td>Phishing</td>
<td>Obtaining sensitive information from victims, or</td>
<td>Sensitive information theft and malware installation.</td>
</tr>
<tr>
<td>Web Attacks</td>
<td>SQL Injection</td>
<td>Targeting vulnerabilities in web applications that rely</td>
<td>Unauthorized access to data and database manipulation.</td>
</tr>
<tr>
<td></td>
<td>XSS</td>
<td>on SQL databases.</td>
<td>Sensitive information theft and user action</td>
</tr>
<tr>
<td>Malware</td>
<td>Malware</td>
<td>Spreading malicious software such as viruses, Trojans,</td>
<td>Sensitive data theft, system disruption, and financial</td>
</tr>
<tr>
<td></td>
<td></td>
<td>and ransomware.</td>
<td>losses.</td>
</tr>
</tbody>
</table>

intentional or malicious actions. Early detection of potential network threats allows for a rapid response, minimizing the risk of intentional harm, service degradation, and monetary loss. AI/ML techniques are recognized as essential enablers for next-generation networks to identify anomalous traffic patterns that could lead to service unavailability or security issues. AI/ML models have demonstrated their ability to uncover hidden patterns in large collections of multi-dimensional, time-varying data [13]. In ZTNs/6G networks, Intrusion Detection Systems (IDSs) should be automatically developed, provisioned, and managed to detect various cyber-attacks.

In the first stage, the ML-based network monitoring system should enable real-time monitoring of all network data from diverse sources, such as connections, devices, radio networks, current and old core networks, services, transport, and Information technology (IT) operations. Monitoring and collecting network data from a variety of sources enables network security managers to gain a multi-perspective view of the network, which is necessary for identifying anomalies that impact multiple domains or environments [22].

Apart from a comprehensive overview of the network, effective network monitoring systems should provide completely autonomous functionalities capable of continuously and optimally adapting to changes and variabilities. By monitoring the complete scope of data and implementing adaptable ML algorithms, cyber-attacks or network abnormalities are identified more quickly and accurately [8].

2) Root Cause Analysis: Root cause analysis aims to identify the underlying cause of network issues, enabling rapid recovery. By analyzing the entire context and relevant information of the detected abnormal events, this procedure aims to determine the possible causes and appropriate countermeasures/solutions for network anomalies. Root cause analysis often requires a comprehensive correlation analysis across multiple architectural levels, events, environments, and vendors [22].

Once an anomaly alert has been triggered, it is crucial to determine its root cause. This is necessary to enable self-organizing systems, implement effective mitigation methods, perform network forensics, and assign responsibility. However, with the complexity and diversity of evolving mobile networks, along with the growing number of Key Performance Indicators (KPIs) and data related to end-users, services, and networks, identifying the underlying cause can be challenging. Manual root cause analysis based on expert knowledge is difficult, time-consuming, and labor-intensive. In this context, AI/ML models have emerged as an attractive alternative for facilitating self-root-cause analysis due to their ability to analyze large amounts of data, identify complex non-linear correlations within the data, and make faster and more accurate decisions compared to manual analysis [13].

3) System Remediation: Network system recovery refers to the process of restoring a network system to its normal functioning state after being compromised by an attack. This is an important aspect of network security, as it helps organizations quickly respond to security incidents and minimize the impact of network attacks [27]. ML-based network monitoring systems pave the way for autonomous remediation by autonomously identifying network anomalies and analyzing their root causes. This information can then be used by the remediation engine to recommend actions for autonomous remediation. Additionally, the remediation operations can be continuously enhanced through fine-tuning in a closed feedback loop.

There are several methods that can recover or remediate a compromised network, including backup and restore, system reimage, network isolation, and blacklisting [22] [28]. The backup and restore method involves creating regular backups of network systems and restoring them after an attack to recover lost or corrupted data. System reimage involves completely reinstalling the operating system and software on compromised network devices, which proves effective in cases where the attack has caused significant damage to the network system. Network isolation is the process of isolating the compromised network segment from the rest of the network using network segmentation techniques to limit the
**TABLE II**
**THE SECURITY CHALLENGES IN ZTNs/6G NETWORKS.**

<table>
<thead>
<tr>
<th>Threat Category</th>
<th>Threat Examples</th>
<th>Description</th>
<th>Countermeasures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open API-based Threats</td>
<td>Parameter Attacks</td>
<td>Security threats targeting APIs in ZSM frameworks due to their increasing importance in network operations.</td>
<td>Authorization, communication encryption, input validation, AI/ML techniques for API threat detection and automated threat mitigation, etc.</td>
</tr>
<tr>
<td></td>
<td>Identity Attacks</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MITM Attacks</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DDoS Attacks</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Intent-based Threats</th>
<th>Information Exposure</th>
<th>Security vulnerabilities in IBN affecting efficiency, flexibility, and security.</th>
<th>Authentication and authorization, secure transport protocol, intent validation, conflict resolution, etc.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Undesirable Configuration</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Abnormal Behavior</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Closed-Loop Network Automation-based Threats</th>
<th>MITM Attacks</th>
<th>Security threats in closed-loop network automation due to the use of AI/ML techniques for self-optimization.</th>
<th>Secure communication channels, AI/ML techniques for real-time CLA attack detection, validation, integrity tests, etc.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Deception Attacks</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Programmable Network Technology-based Threats</th>
<th>SDN Threats</th>
<th>Security threats targeting SDN and NFV technologies in programmable networking solutions.</th>
<th>Authorization, confidentiality and integrity of messages, traffic monitoring and filtering, secure communication channels, etc.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NFV Threats</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

spreads of the attack. This helps prevent further damage and provides a safe environment for recovery efforts. Blacklisting is a common defense method that blocks traffic or requests from detected malicious or suspicious network devices/nodes until further authentication to prevent ongoing attacks. Among AI/ML techniques, Reinforcement Learning (RL) models have been used as a potential autonomous solution to effectively select and recommend actions for fast network recovery [29].

4) **Attack/Intrusion Prevention:** Intrusion prevention is a critical security procedure aimed at preventing unauthorized or malicious access to computer systems and networks. It is an essential aspect of network security, enabling organizations to prevent and respond to security incidents. Both system remediation and intrusion prevention methods belong to cyber-attack countermeasures in ZTNs/6G networks.

There are several methods for intrusion prevention, including firewalls, access control or authentication, encryption, network segmentation, and antivirus software [30] [31] [32]. Firewalls and authentication mechanisms serve as the first line of defense against network attacks. Firewalls can be configured to block incoming traffic based on predefined rules and prevent unauthorized access to the network. Access control or authentication is a security measure aimed at controlling access to a network or specific network resources, such as usernames and passwords, biometric authentication, and multifactor authentication [31]. Encryption or cryptography methods are used to protect sensitive information from unauthorized access. Encrypting sensitive data can prevent attacks that target this data. Network segmentation is an approach that aims to divide a large network into smaller, more secure subnetworks. This can help prevent the spread of network attacks and reduce the attack surface of the network [32]. Antivirus software is specifically designed to identify and remove malware from computer systems and networks. Regularly updating antivirus software can help prevent malware attacks and maintain network security.

**IV. ZTN/6G NETWORK SECURITY THREATS AND POTENTIAL SOLUTIONS**

To ensure the cybersecurity of ZTNs, it is essential to implement effective E2E security management. The increasing number of connected devices and the expanding networks has resulted in a rapid increase in threats targeting 5G/6G networks. The threat surfaces of the ZSM architecture include open API threats, intent-based threats, closed-loop network automation-based threats, and programmable network technology-based attacks [3] [13]. This section discusses the primary security challenges specific to ZTNs, the ZSM architecture, and the corresponding potential counterattack mechanisms. The challenges presented in this section are summarized in Table II. Moreover, the popularity of AI/ML models in enabling autonomous services in ZTNs has introduced a special challenge in the form of adversarial attacks, which aim to compromise AI/ML models and disrupt ZTN services. This challenge will be further addressed in Section V-D.

A. **Open API-based Security Threats and Potential Solutions**

APIs are essential components for integrating web-based applications and facilitating communication within the ZSM framework, which plays an integral role in service provisioning, management, orchestration, and monitoring [33] [34]. As 6G networks follow the trend set by 5G networks, open APIs are expected to play an even more significant role and be further enhanced [1]. However, as APIs are vital to network operations, they also become attractive to cyber-attackers, posing various security threats to ZTNs [3].

Cyber attacks targeting open APIs can be divided into four major categories: parameter attacks, identity attacks, MITM attacks, and DDoS attacks [1] [3] [11]:

i) **Parameter Attacks:** Parameter attacks take advantage of data transmitted through the API, such as query parameters, Hypertext Transfer Protocol (HTTP) headers, Uniform Resource Locators (URLs), and post content. Examples of parameter attacks include script insertions, SQL injections, buffer overflow attacks, and injection attacks targeting the common data services component of the ZSM framework [3]. These attacks can result in unauthorized data access, data manipulation, and DoS attacks.

ii) **Identity Attacks:** Identity attacks exploit vulnerabilities in the authentication and authorization procedures of
ZTNs, such as using extracted API keys as credentials or delivering large quantities of random data to identify vulnerabilities [1]. Insecure APIs can result in unauthorized access to domain orchestration services or E2E service orchestration services, allowing attackers to modify configurations or create E2E service instances to exhaust network resources [3].

iii) MITM Attacks: MITM attacks occur when cyber-attackers intercept unencrypted API messages, capture confidential information, or manipulate messages [11]. These attacks can compromise the confidentiality and security of transmitted data.

iv) DDoS Attacks: APIs are also vulnerable to DoS and DDoS attacks, where attackers can overwhelm an API with a large number of requests, resulting in service unavailability [3].

API security is essential to restrict interactions with ZSM APIs to authorized users [35]. Several security measures, such as authorization, communication encryption, and input validation, can be implemented in ZTNs to mitigate API security threats [11]. The enhancement of authorization can be achieved through the application of Access Control Lists (ACLs), Role-Based Access Control (RBAC), and Attribute-Based Access Control (ABAC). These mechanisms are designed to aid in fulfilling the imperative of the least privilege [11]. To prevent MITM attacks, API messages must be encrypted, and Transport Layer Security (TLS) can be used to improve API message integrity and confidentiality. Validation of all input data in the URL, query parameters, HTTP headers, JavaScript Object Notation (JSON) schema, and content of ZSM APIs is essential for preventing injection attacks. Moreover, AI/ML techniques can be used to improve existing security mechanisms in ZTNs, such as automated API threat detection and mitigation.

B. Intent-based Threats and Potential Solutions

IBN offers a promising approach to integrate AI/ML techniques into 6G networks, addressing the limitations of traditional networks in terms of efficiency, flexibility, and security [11]. By leveraging AI/ML techniques, IBN directly translates users’ business intentions into network configurations, operations, and maintenance strategies. As part of the ZSM architecture, intent-based interfaces provide high-level abstractions and specify policies, enabling the decoupling of technology- and vendor-specific details [34]. However, IBN introduces security vulnerabilities, including information exposure, undesirable configuration, and abnormal behaviors [11].

i) Information Exposure: Intent-based interfaces may expose confidential information about the application’s desires, such as connecting with peers, advertising services or content, and controlling network traffic [11]. This information can be intercepted without authorization, compromising system security objectives (e.g., privacy and confidentiality) and opening the backdoor to additional attacks [3].

ii) Undesirable Configuration: Malicious modification of an intent-based service model may result in a configuration of an undesirable security level, leaving the network segment vulnerable to security threats [11]. For instance, a cyber-attacker could alter a high-security-level intent to a low-security-level intent, resulting in deficient security measures in a network segment.

iii) Abnormal Behavior: A malformed intent sent to a domain orchestration service may trigger abnormal behavior, including service disruption or a DoS attack [11]. Alternatively, the service may transfer the intent to a policy, resulting in a misconfigured network that causes network outages or security breaches.

To mitigate these intent-based threats, several potential mechanisms can be employed [11]:

i) Authentication and Authorization: OpenID Connect, signed JSON Web Tokens (JWT), and OAuth2.0 can ensure mutual authentication between intent producers and consumers and restricted access to intent-based interfaces.

ii) Secure Transport Protocol: The exchange of intents should occur over a secure transport protocol, such as TLS 1.2, to maintain intent integrity and confidentiality, thereby preventing intent tampering and eavesdropping.

iii) Intent Validation and Conflict Resolution: The intent engine should be able to validate the intent format and detect/resolve potential conflict situations, mitigating the impact of malformed or conflicting intents on insufficient network operation and security.

Intent-based interfaces can contribute to the overall security and efficacy of ZTNs within the context of 6G mobile networks by addressing these security concerns and implementing the suggested mitigation mechanisms.

C. Closed-Loop Network Automation-based Threats and Potential Solutions

6G networks will leverage closed-loop network automation for zero-touch management capabilities [36]. Closed-Loop Automation (CLA) continuously monitors, identifies, adjusts, and optimizes network performance to enable self-optimization [1]. As 6G technologies evolve, external CLA capabilities become necessary for addressing the expanding threat landscape. Using AI/ML techniques, CLA security mechanisms have the potential to autonomously detect and rapidly mitigate threats.

However, several security threats are driven by closed-loop networked automation, such as MITM, deception attacks, and other unknown threats [3] [11]. In MITM attacks, an intruder intercepts communications between two parties to eavesdrop or manipulate traffic remotely. The cyber-attacker obtains user credentials and confidential information, and they may interrupt message exchanges, resulting in data corruption. Deception attacks rely on manipulating the target to act based on false information, convincing them to perceive an incorrect version of the truth as fact, and causing the propagation of false feedback information.

To address security threats in CLA, implementing secure communication channels between network components can reduce the risk of MITM and deception attacks and prevent
unauthorized access [11]. Additionally, AI/ML techniques can be used to monitor network traffic and detect anomalies for real-time CLA attack detection. Validation and integrity tests at various phases of a closed-loop process can also assist in preventing the injection and manipulation of malicious data.

D. Programmable Network Technology-based Threats and Potential Solutions

ZTNs leverage SDN and NFV technologies to construct programmable networking solutions [3] [15]. Despite their potential benefits, these technologies introduce new attack vectors and security risks that compromise the application, data, and control layers of SDN [37].

1) SDN Threats and Potential Solutions: The Open Network Foundation (ONF) has identified various threats to SDN, including spoofing, tampering, repudiation, information leakage, DoS, and privilege escalation [11]. These threats can target various SDN layers (i.e., the control, data, and application layers) and often result from design and implementation limitations in existing SDN controller platforms [38]. The risks are increased when users have programmatic access to SDN, particularly when they use third-party applications or standard-based solutions to access the network. Insufficient isolation can lead to cyber-attacks such as network manipulation and Address Resolution Protocol (ARP) spoofing [3] [15]. Several solutions have been suggested for mitigating SDN-related attacks [11] [38]. These include:

i) Implementing authorization mechanisms such as Role-Based Access Control (RBAC) to regulate access levels and prevent privilege abuse.

ii) Ensuring the confidentiality and integrity of messages exchanged on communication channels, both in the North-South and East-West interfaces, through the use of encryption, digital signature, and Message Authentication Code (MAC) algorithms.

iii) Utilizing tamper-proof devices like Trusted Platform Modules (TPM) to secure sensitive data, encryption keys, passwords, and certificates.

iv) Implementing a distributed SDN controller architecture to defend against DoS/DDoS attacks and ensure system availability.

2) NFV Threats and Potential Solutions: NFV is susceptible to various virtualization and networking threats, as well as threats resulting from their combination [39]. These threats include generic virtualization threats like memory leakage and interrupt isolation, generic networking threats like flooding attacks and routing security issues, and threats that arise from the integration of virtualization and networking, such as introspection attacks. Virtual Network Functions (VNFs) are vulnerable to design, implementation, and configuration weaknesses resulting in inappropriate monitoring of data that misleads ZSM service intelligence and E2E analytics [3]. The NFV Security Group of ETSI has proposed various potential security mechanisms to counter NFV-related threats [11] [39]:

i) Trusted Platform Module (TPM) and Virtual TPM: These are recommended for ensuring the integrity of VNF booting and enabling remote verification.

ii) Trusted Execution Environments: Examples include Intel’s Software Guard Extensions (SGX) and Secure Encrypted Virtualization (SEV) for Advanced Micro Devices (AMD). These are utilized for establishing the confidentiality and integrity of VNFs at runtime.

iii) Traffic Monitoring and Filtering: Tools such as IDSs and firewalls are used for detecting DoS/DDoS attacks.

iv) Robust Authentication and Authorization Mechanisms: These are crucial for preventing NFV Management and Orchestration (MANO) hijacking.

v) Secure Communication Channels: These are recommended for preventing MITM attacks.

vi) System Hardening Techniques: These can help prevent the exploitation of software vulnerabilities in NFV components.

V. AI/ML Models and Their Security

A. AI/ML Techniques

AI/ML techniques are becoming increasingly vital for strengthening 6G security. Traditional security solutions, such as firewalls and conventional IDSs, struggle against complex cyber-attacks within a 6G environment. AI/ML techniques have proven to be a viable solution for enhancing security across multiple network layers, spanning from the physical layer to the network layer [18].

At the physical layer, AI/ML significantly bolsters security defenses by improving the performance of detection engines, mitigating vulnerabilities such as eavesdropping and jamming attacks. On the network layer, AI/ML has been leveraged in several key technologies to enhance system performance within the 6G landscape. These include verifying node behavior to detect insider attacks, predicting potential network attacks, optimizing radio and computing control policies, determining the prioritization of equipment recovery, and scrutinizing traffic or network access behavior [18].

AI/ML plays a crucial role in enabling efficient ZTN management within 6G networks by automating and optimizing a wide range of network operations. ML techniques can be categorized into four groups, each with its specific applications in the context of ZTN services functionalities [14] [40].

1) Supervised Learning: Supervised Learning (SL) techniques are frequently utilized in ZTNs for the automation of network and service management [41]. These techniques are especially effective in classification and regression problems, where the objective is to predict a specific class or output value given a particular input. SL is employed for tasks such as traffic classification, service requirement prediction, and user behavior forecasting. There are many different types of SL algorithms, such as linear regression, logistic regression, decision trees, random forests, XGBoost, LightGBM, K-Nearest Neighbors (KNN), Support Vector Machines (SVM), naive Bayes, and Neural Networks (NN) [8] [42] [43].

2) Unsupervised Learning: Unsupervised Learning (UL) techniques are essential in understanding and organizing data without the need for labeled examples. They are used in ZTNs to group network traffic flows with similar characteristics and assign them to appropriate network slices. This
ensures optimal utilization of network resources and allows for differentiated levels of Quality of Service (QoS). Commonly employed UL algorithms for these purposes encompass K-means, Gaussian mixture models, and Principal Component Analysis (PCA) [44] [45].

3) Deep Learning: Deep Learning (DL) techniques, based on artificial neural networks, address the challenges posed by the necessity for manual feature extraction in traditional ML approaches [46] [47]. They enable the efficient processing of raw natural data and facilitate the implementation of large state-space actions. Common DL algorithms include Multi-Layer Perceptron (MLP), Convolutional Neural Networks (CNNs), and Recurrent Neural Networks (RNNs) [44]. In the context of ZTNs, DL algorithms such as deep Q-learning and Soft Actor-Critic (SAC) are used for tasks like traffic classification and prediction. Radio Access Network (RAN) management, multi-domain orchestration, and resource management.

4) Reinforcement Learning: RL techniques enable adaptive decision-making in network management by modeling problems as driven by rewards and punishments [48]. They find use in various applications, including slice admission strategies, data migration in MEC, Radio Access Technology (RAT) selection, and resource allocation for network services [44]. RL algorithms are categorized as model-based and model-free models, and examples such as proactive resource allocation and Q-learning are utilized in ZTN scenarios.

B. Challenges of Traditional AI/ML Techniques

Despite the widespread use of traditional ML algorithms for various cybersecurity tasks, they encounter several critical challenges [7] [8] [9].

i) Time-Consuming Model Development: Traditional ML and data analytics techniques involve a sequence of manual tasks, such as data pre-processing, feature engineering, model selection, and hyperparameter tuning. These tasks require significant time and effort from ML developers and researchers, which can impede the rapid development and deployment of ML models within 5G/6G networks.

ii) Human Bias and Errors: The inherent subjectivity involved in manual ML model development can introduce potential biases and errors. These can result in issues such as overfitting and underfitting, leading to a degradation in model performance.

iii) Expertise Requirements: The deployment of traditional ML algorithms in practical applications often necessitates a high level of expertise. This can create a barrier to interdisciplinary collaboration by increasing the demand for skilled ML professionals.

iv) Insufficient Adaptability: Traditional ML models are often slow or unable to react to ever-changing data streams in dynamic networking environments. They generally lack the ability to detect unknown or zero-day attacks.

While traditional ML algorithms present certain challenges, they remain vital techniques in the pursuit of enhanced security within the landscape of 6G networks. By addressing these challenges, we can further leverage the potential of AI/ML in cybersecurity, improving the resilience and adaptability of our future networks.

C. Network Automation Solutions: Automated Machine Learning (AutoML)

Although ML algorithms can learn data without human intervention, there are certain procedures in ML pipelines that require experienced data scientists. These procedures include preparing appropriate and sanitized data, selecting the most suitable ML algorithm, tuning hyperparameters, and determining whether the ML model needs updating. Data scientists usually experiment with various pre-processing methods and ML algorithms to find the optimal combination. These procedures are human-centric, time-consuming, and require specialized expertise in ML and data analysis [8]. Automating the process of designing and tuning ML methods is known as AutoML. AutoML refers to the fully automated process of applying ML algorithms to real-world practical applications, allowing both beginners and experts to develop ML models in a more efficient way.

AutoML aims to enable domain specialists to create ML models automatically without requiring extensive knowledge and experience in ML. AutoML addresses the shortage of data scientists and has the potential to significantly improve the performance and effectiveness of ML models by shortening work cycles and increasing model performance [8]. Given the high demand for network automation, AutoML techniques have become necessary components in ZTNs/6G networks.

ZTNs represent autonomous networks with self-healing and self-protection capabilities based on the data collected and analyzed across all network activities [11]. Robust and powerful AI/ML models are key components in safeguarding ZTNs by transforming conventional network security operations into automated and intelligent operations. In the general cybersecurity process discussed in Section III-B, supervised and unsupervised AutoML techniques can be developed to automatically detect malicious cyber-attacks. RL-based AutoML techniques can then be used to automatically determine appropriate actions to defend against cyber-attacks and aid in system remediation.

Figure 2 illustrates the general AutoML framework, which consists of four stages: automated data pre-processing, automated feature engineering, automated model learning, and automated model updating. The specifics of each stage of the AutoML pipeline are described below.

1) Automated Data Pre-Processing: Data pre-processing enhances data quality for ML model development by addressing common issues such as outliers, missing values, and class imbalance [49]. Automated Data Pre-processing (AutoDP) is an essential component of AutoML, streamlining the time-consuming and tedious pre-processing stage. AutoDP tasks can be categorized into four main areas: data transformation, imputation, balancing, and normalization [8] [50].

i) Data Transformation: It involves encoding techniques for converting categorical features into continuous ones (e.g., label encoding and one-hot encoding) and discretization.
Fig. 2. An overview of the general AutoML framework.

Techniques for transforming continuous features into categorical ones [8].

ii) Data Imputation: It addresses the issue of missing values commonly found in real-world network datasets, which can significantly impact the learning performance of ML models. Common data imputation approaches include zero/mean/median imputation for numerical and categorical features, forward/backward filling for time-series data, and ML model-based imputation techniques for advanced imputation [51].

iii) Data Balancing: It addresses class imbalance issues in network datasets for classification problems, which can result in ML model performance degradation. Resampling techniques, such as over-sampling techniques (e.g., the Synthetic Minority Over-sampling Technique (SMOTE) [21] and ADaptive SYNthetick (ADASYN) [52]) and under-sampling methods (e.g., random under-sampling), can be employed to mitigate class imbalance by generating minority class samples or removing majority class samples [53].

iv) Data Normalization: It ensures fair treatment of features with different scales by ML models, especially when distance calculations are involved in model learning. Z-score and min-max normalization are two common approaches used in ML model learning, with the choice depending on the occurrence and percentage of outliers [54].

2) Automated Feature Engineering: Feature Engineering (FE) is a critical aspect of the ML pipeline that transforms or creates features from existing data to improve data quality. FE can determine the upper limit of ML applications [9]. However, manual FE is laborious and requires domain knowledge. To address this issue, Automated FE (AutoFE), an important AutoML procedure, automates the generation and selection of relevant features, making the process more efficient and reproducible [8]. AutoFE combines three main components: feature generation, feature selection, and feature extraction [8]. The generate-and-select strategy is employed by state-of-the-art AutoFE approaches to automate the traditional FE process, such as AutoFeat. The general AutoFE process involves creating an exhaustive feature pool through feature generation methods and selecting or extracting valuable features using feature selection and extraction methods [55].

i) Feature Generation: It aims to create new features by transforming or combining existing features to improve the generalizability and robustness of ML models. Common feature generation operations include unary operations (e.g., logarithmic operations), binary operations (e.g., addition, subtraction, multiplication, and division), and high-order operations (e.g., maximum, minimum, and average) [56]. For network time-series data, features such as date/time features, lag features, and window-based features can be generated to improve model performance.

ii) Feature Selection: It aims to identify the most appropriate features and remove irrelevant or redundant ones. Automated feature selection, or AutoFS, frames the feature selection problem as an optimization problem to automatically determine the optimal number or percentage of selected features [57]. Existing feature selection methods can be categorized into filter methods (e.g., Information Gain (IG), and Pearson correlation), wrapper methods (e.g., Recursive Feature Elimination (RFE)), and embedded methods (e.g., Lasso regularization and decision tree-based algorithms) [8] [21] [58] [59].

iii) Feature Extraction: It aims to reduce dimensionality by applying mapping functions, creating a more concise representation of the original dataset while potentially increasing model learning efficiency [59]. Common feature extraction methods include Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Autoencoders (AE) [9]. Feature extraction is typically utilized when the feature set remains highly dimensional or under-performing after feature generation and selection.

In summary, AutoFE is a dynamic combination of feature generation, feature selection, and feature extraction to automate the FE process for improved efficiency and reproducibility. More accurate and efficient ML models can be developed.
by selecting the appropriate AutoFE methods and optimization techniques, ultimately advancing network performance.

3) Automated Model Learning: The primary challenge in applying ML algorithms to real-world tasks resides in the selection and configuration of the most appropriate ML model to achieve optimal results. For a prediction task, the accuracy of different models with different configurations may vary significantly [49]. As a result, it is crucial to identify the most suitable ML model with optimal hyperparameter configuration.

Hyperparameters are the parameters of ML algorithms that define the architecture of ML models and must be specified before model learning [44]. According to the domains of hyperparameters, they can be classified as continuous hyperparameters (e.g., the learning rate of neural networks), discrete hyperparameters (e.g., number of nearest neighbors in KNN), and categorical hyperparameters (e.g., the kernel type in SVM). Additionally, certain hyper-parameter configurations include conditioning. For example, in neural networks, the number of hidden layers and the number of neurons in each layer have strong correlations [44]. Conditional hyperparameters must be tuned together to identify the optimal configuration.

Selecting an appropriate ML algorithm and hyperparameter values can be seen as a search problem. The set of potential ML models and their corresponding hyperparameter combinations constitute a search space, wherein each ML model, along with its specific hyperparameter configuration, represents a point within this space. Detecting the optimal point in the search space is a global optimization problem [44]. Therefore, using optimization techniques to automatically detect the optimal ML algorithm and hyperparameter configuration is defined as Combined Algorithm Selection and Hyper-parameter (CASH) optimization problems [62]. CASH systems consist of two levels: automated model selection and HPO. At the first level, the suitable ML models are selected based on their default hyperparameters. At the second level, model-specific hyperparameters are optimized to obtain the final model with the best performance [49].

Generally, CASH is defined as identifying the most suitable ML model $A^*$ with its optimal hyperparameter configuration $\lambda^*$ that minimizes the loss function $L$, as depicted by the following equation:

$$A^*, \lambda^* \in \arg \min_{A^{(j)} \in A, \lambda^{(j)} \in \lambda} \frac{1}{K} \sum_{i=1}^{K} L \left( A^{(j)}, \lambda^{(j)}, D_{\text{train}}^{(i)}, D_{\text{valid}}^{(i)} \right)$$

where $A^{(j)} \in A$ is the ML algorithm to be selected, $\lambda^{(j)} \in \lambda$ refers to the hyperparameters to be optimized, $D_{\text{train}}$ and $D_{\text{valid}}$ are the training and validation datasets, and $K$ indicates the number of folds in k-fold cross-validation.

Similarly, Neural Architecture Search (NAS) is the process of automating the design of DL models [7]. Apart from the hyperparameters in DL models, such as the learning rate, NAS techniques focus on automatically determining the optimal neural network architecture, such as the number of hidden layers, the type of each layer, and the number of neurons in each layer.

Optimization techniques are used to solve CASH and NAS problems by automatically detecting the most suitable ML model or DL architecture with the best-performing hyperparameter configuration as the optimal solution [44]. Common optimization techniques for CASH and NAS tasks include Grid Search (GS), Random Search (RS), Bayesian Optimization with Gaussian Process (BO-GP), Bayesian Optimization with Tree Parzen Estimator (BO-TPE), Hyperband, Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and RL [8]. Table [11] provides a summary of these optimization techniques, outlining their descriptions, strengths, and limitations. Suitable optimization methods can be selected for specific tasks according to their respective strengths and limitations.

4) Automated Model Updating: After the automated model learning process to train optimal ML models, these models are deployed into real-world production environments, leveraging their capabilities to solve complex tasks. However, when these ML models are introduced into dynamic networking systems, they may encounter substantial performance degradation due to model drift issues [10].

Model drift refers to the decline in the ML model’s performance over time and can be further classified as data drift and concept drift. Data drift refers to the unpredictable changes in data distributions, while concept drift refers to the shifts in the relationships between the input features and the target variables [61]. In cybersecurity applications, changes in the benign event data distributions are examples of data drift, while zero-day attacks exemplify concept drift [8]. Both data and concept drift can lead to the degradation of ML model performance, thereby underscoring the necessity for automated model updating. This process aims to preserve high prediction accuracy in non-stationary networking environments [8]. This process aligns with continual learning, a crucial ML procedure for data stream analytics.

To handle model drift, learning systems must promptly detect and adapt to these changes. Drift detection and adaptation methods are essential components of automated model updating designed for streaming data with data or concept drift issues. Drift detection methods are primarily classified into two categories [62]:

i) Distribution-based Methods: They detect drift by monitoring data drift or data distribution changes in time windows. Techniques such as ADaptive WINdowing (ADWIN) [63], Information Entropy (IE), and Kullback-Leibler (KL) divergence [64] are commonly used for distribution-based drift detection. These methods are often used in systems with limited memory and can provide high interpretability. However, they may incur higher computational costs than performance-based methods.

ii) Performance-based Methods: They track changes in performance or prediction error rates of ML models to identify model drift. Popular methods like Drift Detection Method (DDM) and Early Drift Detection Method (EDDM) use pre-defined thresholds to determine the occurrence of drift [65]. Performance-based methods are particularly effective for detecting sudden drift in data streams but may have limitations in identifying gradual drift.
Once drift is detected, adapting to the drift becomes crucial to update the ML models and maintain their performance in the context of data streams. There are three primary types of drift adaptation methods:

i) **Model Retraining**: Model retraining is a simple approach for drift adaptation that involves updating learning models on newly arrived data streams. Strategies include full retraining, partial retraining, and instance weighting. The full retraining process entails retraining the learning model on all available samples in the dataset. Partial retraining methods, such as OASW [10], use adaptive or sliding windows to detect model drift and collect new concept samples, allowing for efficient model updating. Instance weighting adjusts the weights of data samples based on their recency, which helps the learning model adapt to model drift by retraining on weighted samples [66].

ii) **Incremental Learning Methods**: Incremental learning methods continuously update learning models sequentially as new data samples are introduced, ensuring the models stay up-to-date and adapt to evolving data. Hoeffding Trees (HT), Adaptive Online Neural Networks (AONN), and their variants are popular incremental learning methods. HT algorithms, such as Very Fast Decision Tree (VFDT), Concept-adapting VFDT (CVFDT), and Extremely Fast Decision Tree (EFDT), use Hoeffding’s inequality to partially update nodes as new samples arrive, making them suitable for data stream analytics due to their ability to adapt to new samples [67]. AONN is another incremental learning method based on neural networks that updates the model when the error rate increases [68]. However, it’s worth noting that while incremental learning methods offer benefits for adapting to new data, they are not specifically designed to address model drift issues.

iii) **Ensemble Learning Methods**: Ensemble learning techniques combine multiple base learners to achieve improved model drift adaptation performance. Block-based ensembles and online ensembles are two main categories of ensemble methods for data stream analytics. Block-based ensembles, such as Streaming Ensemble Algorithm (SEA) [69], Accuracy Weighted Ensemble (AWE) [70], and Diversity and Transfer-based Ensemble Learning (DTEL) [71], train base learners on discrete data blocks and update the ensemble when new blocks are added to the database. Online ensembles, such as Adaptive Random Forest (ARF) [72], Streaming Random Patches (SRP) [74], and Performance Weighted Probability Averaging Ensemble (PWPAE) [65], integrate multiple incremental learning models to enhance continual learning performance. Ensemble learning models are generally efficient for handling gradual and recurring drifts but may struggle with abrupt drifts and increased computational complexity.

In summary, the detection and adaptation of model drift are essential steps for automated ML model updating to maintain high model performance in dynamic ZTNs/6G networks.

### D. Adversarial ML Attacks

The automation of modern networks has experienced significant improvement through the application of AI/ML tech-
techniques, resulting in enhanced functionalities such as self-planning, self-optimization, self-healing, and self-protection [17]. However, AI/ML techniques introduce new attack interfaces in ZTNs/6G networks. The AI/ML models are vulnerable to attacks during the training and testing phases, compromising the integrity, availability, and privacy of the system [74]. These targeted cyber threats against AI/ML models are known as AML attacks [75]. Although AML attacks are not explicitly designed for ZTNs/6G networks, the increasing demand for network automation and reliance on AI/ML techniques has elevated AML attacks to a significant threat in these networks. Due to insufficient human supervision, it is challenging for 5G/6G networks to maintain AI/ML model security. Additionally, given the capabilities and extensive uses of AI/ML methods, cyber-criminals can devise complex and advanced cyber-attacks with potentially severe and harmful consequences [13].

In AML attacks, cyber-attackers aim to deceive an AI/ML model by feeding it carefully crafted data intended to trigger inaccurate predictions or classifications. AML attacks can be classified based on the stage at which they occur during the ML process, including data poisoning attacks, evasion attacks, and inference attacks [13] [76]. Data poisoning attacks occur during the training stage, where the cyber-attacker injects malicious data into the training set to contaminate the model. Evasion attacks occur after the ML model has been trained, where the cyber-attacker manipulates the model to degrade its performance. Inference attacks occur during the deployment/prediction phase, where the cyber-attacker manipulates the prediction data given to the ML model to cause a specific outcome and potentially gain access to sensitive information. Figure 3 illustrates these three types of AML attacks and potential defense mechanisms. The specifications for each type of attack are discussed below.

1) Poisoning Attacks: In poisoning attacks, an attacker manipulates data or the AI/ML model during the training phase to influence the prediction results [75]. Attackers can exploit the need for continuous retraining learning models to adapt to new data distributions, thereby creating an opportunity to maliciously manipulate or poison the trained model. In classification problems, poisoning attackers alter the labels of data samples to cause the ML model to misclassify them, while in regression problems, they modify the values of input features to cause the model to produce incorrect outputs [77]. Poisoning attacks can be executed using various strategies, such as data injection, data manipulation, and logic corruption [13] [75].

In data injection attacks, the attackers do not have access to the training data and aim to alter the data distribution by introducing crafted malicious samples into the training dataset while keeping the original samples unchanged. This method is effective when the original classifier is trained with limited data and requires additional data for retraining [76].

Data manipulation attacks, on the other hand, assume that attackers have full access to the training data, allowing them to directly contaminate the original data used for training the learning model [13]. Contamination involves changing labels (e.g., malicious to benign) or adding small perturbations to the input features. Attackers often target samples or labels with high classification confidence to maximize their impact on the ML model while minimizing the chances of detection [75].

Logic corruption attacks specifically target the learning algorithm or its underlying logic with the intention of disruption. Logic corruption can be utilized against models that employ distributed learning, such as federated learning, which relies on the training of multiple agents. In such attacks, a malicious agent can manipulate the local model parameters to compromise the global model [13].

Poisoning attacks can cause the classifier to make erroneous judgments, resulting in reduced accuracy, changes to decision boundaries, and even errors in final test results [78]. Poisoning attacks pose serious risks to network performance, security, and privacy in 6G networks, which rely significantly on ML for numerous applications and services.

2) Evasion Attacks: The purpose of evasion attacks is to discover challenging data samples that are most likely...
to be misclassified to deceive AI/ML models into making incorrect decisions. These attacks, unlike poisoning attacks, do not affect the training process of ML models. Instead, they occur during the testing phase, where the attacker introduces adversarial examples, or minor perturbations, to the input instances [76].

Evasion attacks exploit the sensitivity of high-precision ML models to subtle perturbations, rendering ML-enabled machines in autonomous networks vulnerable to being deceived by adversarial examples [23]. These attacks can result in system misclassification or reduced accuracy, posing privacy risks for ML-based applications in autonomous networks.

Common evasion attacks include Decision Tree Attack (DTA) [79], Fast Gradient Sign Method (FGSM) [80], and Basic Iterative Method (BIM) [81]. DTA targets decision tree-based ML models (e.g., random forest, XGBoost, and LightGBM) by manipulating input features to mislead the models. This attack demonstrates the need for enhanced protections in tree-based classifiers to maintain their reliability in security-critical applications [79]. FGSM generates adversarial samples by introducing imperceptible amounts of noise to the input data using gradients derived from the model’s loss function [80]. Due to its efficiency, FGSM poses challenges to ML models, particularly DL models. Consequently, it calls for the development of countermeasures to defend against gradient-oriented attacks. BIM is an iterative variant of the FGSM attack in which adversarial perturbations are implemented incrementally over multiple iterations [81]. By allowing fine-grained control over the perturbation process, BIM can generate more effective adversarial examples. This iterative approach highlights the importance of robust defenses that can withstand iterative gradient-based attacks.

Evasion attacks constitute the primary category of AML attacks, and cyber-attackers employ various techniques to compromise the performance of AI/ML models. Examples of such evasion attacks include the Jacobian-Based Saliency Map Attack (JSMA), Projected Gradient Descent (PGD), Zeroth-Order Optimization (ZOO) attack, and High Confidence Low Uncertainty (HCLU) Attack [82].

Defending against evasion attacks is challenging due to various factors, including the high dimensionality and continuous nature of the input space, the difficulty in precisely modeling the decision boundary of the ML models, and the restricted availability of labeled data for training. In addition, the effectiveness of evasion attacks often relies on the particular ML algorithm, its settings, and the quality of the training data.

3) Inference Attacks: Inference or exploratory attacks refer to AML attacks where the attacker aims to gain knowledge or confidential information about the target ML model, its input data, or its architecture [76]. These attacks can be especially harmful in security-critical applications, such as privacy-preserving ML models, where the attacker’s goal is to infer sensitive information about individuals or network devices.

Inference attacks include model inversion, model extraction, and membership inference attacks [13]. Model inversion attacks attempt to recover the training data by exploiting the model’s outputs. Model extraction attacks seek to reveal the model’s architecture and parameters to replicate a nearly identical ML model by observing the model’s predictions and/or execution time. Membership inference attacks exploit the model’s output to determine whether a data sample was part of the training dataset used for the target ML model.

Inference/exploratory attacks pose a threat to the privacy and confidentiality of sensitive information contained within the ML models and their training data [78]. The success of inference attacks often depends on the structure and parameters of the ML model, the quality and quantity of the training data, and the presence of privacy-preserving mechanisms, such as differential privacy. In addition, the efficacy of the attack can be affected by the attacker’s knowledge of the ML model and training data.

E. AI/ML Model Security Solutions: AML Defense Mechanisms

AML defense is an emerging research field focused on enhancing the resilience of ML approaches against adversarial attacks [75]. Its primary objective is to assess the vulnerability of ML algorithms to adversarial attacks and develop effective responses that promote more robust learning. Several defense mechanisms have been proposed to address adversarial attacks, such as input validation, adversarial training, adversarial sample detection, defense Generative Adversarial Networks (GANs), concept drift adaptation, differential privacy, and homomorphic encryption [11], as illustrated in Fig. 5.

1) Countermeasures Against Poisoning Attacks: There are several potential countermeasures/solutions to defend against poisoning attacks. Input validation involves the sanitization of training data by eliminating malicious or unusual samples before integrating them into the ML model [13]. Data sanitization and anomaly detection methods are also used as countermeasures to remove suspicious/malicious samples from the training data [77].

2) Countermeasures Against Evasion Attacks: Evasion attacks can be defeated by adversarial training, adversarial sample detection, ensemble methods, and adversarial concept drift adaptation techniques [13] [83]. Adversarial training can train/retain AI/ML models on an augmented dataset with adversarial samples to improve model performance. Adversarial samples can be generated by defense GANs. In adversarial sample detection countermeasures, the adversarial samples generated by AML attacks are detected and removed from the input data. Ensemble learning methods can improve the robustness of AI/ML models and reduce the impact of polluted models by integrating multiple base learners. Adversarial concept drift adaptation techniques can update or retrain AI/ML models to confront AML attacks once a data distribution change or a model performance degradation is detected.

3) Countermeasures Against Inference Attacks: To mitigate the impact of inference attacks, researchers have proposed several approaches, including Differential Privacy (DP), data perturbation, and homomorphic encryption [84]. In DP methods, random noise is injected into the model’s outputs to prevent the attacker from gaining knowledge about the training data or individual samples. Data perturbation aims to perturb
or transform the training data to make it difficult for the attacker to infer sensitive information. Homomorphic encryption enables model training with encrypted data, ensuring the confidentiality of data. However, this approach introduces computational complexities that need to be carefully managed [13].

4) Summary: The common types of AML attacks and their potential countermeasures are summarized in Table IV. Although there are many countermeasures to general AML attacks, cyber-attackers have been continually evolving their tactics to evade these defenses and breach ML models. Therefore, the development of effective defense mechanisms for AML attacks in ZTNs is critical to ensure the security and robustness of ML models integrated into ZTNs/6G networks.

VI. CASE STUDY

With the introduction of ZTNs, AutoML techniques, and AML attacks, this section presents two case studies to illustrate the capabilities and benefits of applying AutoML techniques to ZTNs and depict the typical AML attack and defense process. The first case study presents a detailed AutoML pipeline designed for automated intrusion detection within ZTNs. In the second case study, we simulate three common adversarial attacks and devise basic defense mechanisms to safeguard the IDS model obtained from the first study. The experimental setups and results analysis for each case study are presented in Sections VI-B and VI-C.

A. Use Case Description

With the rise of 5G networks, the increase in connected devices and data traffic has broadened the attack surface, making networks more vulnerable to cyber threats. The evolution towards ZTNs/6G networks necessitates communication systems that are not only more secure but also more efficient and advanced. IDSs, as discussed in Section III-B1, play a pivotal role in monitoring and detecting security threats and malicious activities within modern networks.

However, ZTNs require high levels of automation and minimal human intervention. Therefore, the development of automated IDSs becomes essential to achieve optimal network management in ZTNs/6G networks. As discussed in Section VI-C, AutoML is an advanced ML technology that automates the design and implementation of ML models. This improves traditional ML procedures, making them more automatic and optimized. By leveraging AutoML techniques, it is possible to develop efficient, accurate, and adaptable IDSs that overcome the performance and human supervision limitations of traditional IDSs [85].

Therefore, the first case study aims to provide a comprehensive analysis of the development and implementation of an AutoML-based IDS for ZTNs, particularly in the context of 6G. By leveraging SL algorithms and optimization methods outlined in Sections V-A and V-C, the case study aims to establish optimal classification models capable of distinguishing malicious attacks from benign events.

The second case study focuses on the cyber-defense exercise of AML attacks targeting the AutoML-based IDS derived from the first case study. As many ZTN services and functionalities rely on AI/ML models, they are becoming increasingly vulnerable to AML attacks. AML attacks exploit the weaknesses and vulnerabilities of ML models by generating adversarial inputs that can deceive or manipulate the models into making incorrect predictions [75]. In the context of ZTNs/6G networks, AML attacks pose a significant threat to overall network security and reliability. This case study aims to demonstrate the detrimental impact that AML attacks can have on ML models in ZTNs/6G networks, and presents basic defense strategies to mitigate these attacks, thereby ensuring the accuracy of the ML-based IDS. In this case study, three common types of adversarial attacks discussed in Section V-D2 (i.e., DTA [79], FGSM [80], and BIM [81]) are used to generate adversarial samples to probe the vulnerability of the IDS. Subsequently, the basic defense mechanisms, including adversarial sample detection and filtering, are devised to safeguard the AutoML-based IDS for ZTNs.

Two public benchmark network security datasets are utilized in these case studies to evaluate the proposed AutoML-based IDS and the AML attack & defense models: the Canadian Institute for Cybersecurity Intrusion Detection System 2017 (CICIDS2017) dataset [86] and the 5th Generation Network Intrusion Detection Dataset (5G-NIDD) [87]. The CICIDS2017 dataset is a state-of-the-art intrusion detection dataset with the latest network threats. The CICIDS2017 dataset is close to real-world network data since it has a large amount of network traffic data, a variety of network features (80), various types of attacks (14), and highly imbalanced classes. The primary types of attacks in the CICIDS2017 dataset include DoS/DDoS,
botnets, brute force, port scan, and web attacks.

The 5G-NIDD dataset, created in December 2022, is a fully labeled resource constructed on a functional 5G test network for researchers and practitioners evaluating AI/ML solutions in the context of 5G/6G security [87]. 5G-NIDD encompasses data extracted from a 5G testbed connected to the 5G Test Network (5GTN) at the University of Oulu, Finland. The dataset is derived from two base stations, each featuring an attacker node and multiple benign 5G users. The attacker nodes target a server deployed within the 5GTN MEC environment. The attack scenarios captured in the dataset primarily include DoS attacks and port scans. By employing these two up-to-date datasets that closely mirror real-world network traffic, the case study results become highly relevant to intrusion detection tasks and challenges in ZTNs/6G networks.

The case studies are conducted using a reduced CI-CIDS2017 dataset with 28,307 samples and a reduced 5G-NIDD dataset with 12,159 instances for the purpose of this work. The experiments were run on a machine with an i7-8700 processor and 16 GB of memory, representing ZTN edge server machines. The functions and methods used in the case studies are implemented using several Python libraries: Sklearn [83], Optunity [89], LightGBM [90], and Adversarial Robustness Toolbox (ART) [82].

To evaluate the ML/AutoML models in the case studies, 80%/20% hold-out cross-validation is used in the experiments, as it is a commonly-used split that can help maintain a balance between bias and variance [91]. To gauge the effectiveness of the models, a combination of four classification performance metrics is employed: accuracy, precision, recall, and F1-scores. A model efficiency metric is also utilized in terms of learning time. These four classification metrics are often used in ML research as they provide a well-rounded view of ML model performance [92]. Due to the inherent imbalance in intrusion detection datasets, relying solely on individual performance metrics, such as accuracy, precision, or recall, can lead to a partial understanding of model performance. Therefore, we considered accuracy, precision, recall, and F1-scores collectively for a comprehensive comparison of ML models to avoid skewed evaluation results. The F1-score, which calculates the harmonic mean of recall and precision scores, provides a balanced view of anomaly detection outcomes while reducing bias. By including both false negatives (assessed by recall) and false positives (measured by precision) in its calculation, the F1-score offers a comprehensive performance metric for evaluating the proposed AutoML pipeline.

Furthermore, to cater to the processing time and efficiency demands of network systems, the learning time of each model is analyzed to compare the learning speeds of various ML models. The ideal ML model for network data analytics should strike a balance between effectiveness and efficiency, ensuring optimal performance while minimizing computational overhead.

B. Use Case 1: Automated Intrusion Detection

1) Experimental Setup: The first case study utilizes a comprehensive AutoML pipeline to achieve automated intrusion detection for ZTNs/6G networks. The proposed AutoML pipeline enables the automation of all necessary procedures in ML/data analytics, including automated data pre-processing, automated feature engineering, automated model selection, and hyperparameter optimization.

The AutoML pipeline begins with AutoDP, which involves these steps:

i) Encoding: The pipeline automatically detects categorical features and transforms them into numerical features using label encoding. This enhances the data’s readability for ML models.

ii) Imputation: It automatically spots missing values in the datasets and fills them with the median values using median imputation. This step enhances data quality.

iii) Normalization: The pipeline chooses an appropriate normalization method (z-score or min-max normalization) to bring features to a similar scale. This enhances data quality.

iv) Balancing: The proposed system automatically identifies whether the dataset is imbalanced. If an imbalance is identified, synthetic minority class samples will be generated using the ADASYN method [52] to balance the data and improve data quality.

Next, the pipeline enters the AutoFE stage. In this stage, the RFE method is used to select vital features and discard irrelevant or noisy features. RFE is a wrapper feature selection method that recursively generates and evaluates feature subsets to remove unimportant features until a defined number of features is selected [58]. To automate this step, PSO is used to optimize the number of selected features as a hyperparameter.

After AutoDP and AutoFE, the pipeline enters the automated model learning stage. In the first step of this stage, the pipeline automatically selects learning models from a set of four representative ML models: K-Nearest Neighbors (KNN), Multilayer Perceptron (MLP), Random Forest (RF), and LightGBM. KNN serves a basic and representative low-complexity ML algorithm, while MLP represents a basic DL model commonly used in various applications. RF and LightGBM are two representative ensemble ML models known for effectively handling non-linear and complex network data. Their widespread usage in data analytics applications attests to their high generalizability. After assessing the performance of each learning model in terms of accuracy and F1-scores, both the top-performing model and the second best-performing model with default hyperparameters are selected for further evaluation through HPO. This selection strategy, incorporating both the top and second best-performing models, mitigates the risk of overlooking the true optimal model and increases the likelihood of identifying the most suitable one.

Once the top two learning models are identified, the HPO technique is applied to fine-tune their hyperparameters, yielding two optimized models. From these, the final optimal model is chosen. PSO is selected as the HPO method due to its superior performance in handling large hyperparameter spaces, low complexity, and generalization ability [44].

Overall, the procedures, methods, and aims/operations of each stage of the proposed AutoML pipeline are summarized in Table. [ ]
TABLE V
THE SPECIFICATIONS OF THE PROPOSED AUTOML PIPELINE.

<table>
<thead>
<tr>
<th>Category</th>
<th>Procedure</th>
<th>Method</th>
<th>Aim/Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutoDP</td>
<td>Encoding</td>
<td>Label Encoding</td>
<td>Convert categorical features into numerical features to enhance the readability of the data for ML models.</td>
</tr>
<tr>
<td></td>
<td>Imputation</td>
<td>Median Imputation</td>
<td>Identify and impute missing values with the median values to improve data quality.</td>
</tr>
<tr>
<td></td>
<td>Normalization</td>
<td>Z-Score or Min-Max Normalization</td>
<td>Automatically select a method to normalize features to a similar scale to enhance data quality.</td>
</tr>
<tr>
<td></td>
<td>Balancing</td>
<td>ADA SYN</td>
<td>Generate synthetic minority class samples to balance data and enhance data quality.</td>
</tr>
<tr>
<td>AutoFE</td>
<td>Feature Selection</td>
<td>RFE</td>
<td>Select important features and remove irrelevant features to improve model efficiency.</td>
</tr>
<tr>
<td></td>
<td>Model Selection</td>
<td>Pearson Correlation</td>
<td>Remove redundant features to improve model efficiency and accuracy.</td>
</tr>
<tr>
<td>AutoFE</td>
<td>Model Selection</td>
<td>KNN</td>
<td>Select the best-performing model among four alternative ML models by evaluating their learning performance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MLP</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>RF</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>LightGBM</td>
<td>Tune the hyperparameters of the ML models to optimize them.</td>
</tr>
<tr>
<td>HPO</td>
<td></td>
<td>BO-TPE</td>
<td></td>
</tr>
</tbody>
</table>

TABLE VI
THE EXPERIMENTAL RESULTS OF THE PROPOSED AUTOML-BASED IDS ON THE CICIDS2017 AND 5G-NIDD DATASETS.

<table>
<thead>
<tr>
<th>AutoML Procedures</th>
<th>Learning Algorithm</th>
<th>CICIDS2017 Dataset</th>
<th>5G-NIDD Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Name</td>
<td>Accuracy (%)</td>
<td>Precision (%)</td>
</tr>
<tr>
<td>No</td>
<td>KNN</td>
<td>97.238</td>
<td>92.081</td>
</tr>
<tr>
<td></td>
<td>MLP</td>
<td>88.516</td>
<td>94.277</td>
</tr>
<tr>
<td>AutoDP &amp; AutoFE</td>
<td>KNN</td>
<td>97.058</td>
<td>92.024</td>
</tr>
<tr>
<td></td>
<td>MLP</td>
<td>85.968</td>
<td>92.069</td>
</tr>
<tr>
<td></td>
<td>LightGBM</td>
<td>99.823</td>
<td>99.468</td>
</tr>
</tbody>
</table>

TABLE VII
THE HPO CONFIGURATION OF THE BEST-PERFORMING LEARNING MODELS.

<table>
<thead>
<tr>
<th>Model</th>
<th>Hyperparameter Name</th>
<th>Configuration Space</th>
<th>Optimal Value on CICIDS2017</th>
<th>Optimal Value on 5G-NIDD</th>
</tr>
</thead>
<tbody>
<tr>
<td>RF</td>
<td>n_estimators</td>
<td>[50, 5000]</td>
<td>400</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>max_depth</td>
<td>[3, 50]</td>
<td>25</td>
<td>38</td>
</tr>
<tr>
<td></td>
<td>min_samples_split</td>
<td>[2, 11]</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>min_samples_leaf</td>
<td>[1, 11]</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>criterion</td>
<td>[gini, entropy]</td>
<td>entropy</td>
<td></td>
</tr>
<tr>
<td>LightGBM</td>
<td>n_estimators</td>
<td>[50, 5000]</td>
<td>300</td>
<td>340</td>
</tr>
<tr>
<td></td>
<td>max_depth</td>
<td>[3, 50]</td>
<td>42</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>learning_rate</td>
<td>[0, 1]</td>
<td>0.884</td>
<td>0.101</td>
</tr>
<tr>
<td></td>
<td>min_leaves</td>
<td>[100, 2000]</td>
<td>1400</td>
<td>1300</td>
</tr>
<tr>
<td></td>
<td>min_child_samples</td>
<td>[10, 50]</td>
<td>30</td>
<td>45</td>
</tr>
</tbody>
</table>

2) Experimental Results and Analysis: The effectiveness of the AutoML framework was assessed by comparing the accuracy, precision, recall, F1-score, and model learning time of ML models developed with and without using AutoML techniques. The comparative performance of the ML/AutoML models on the CICIDS2017 and the 5G-NIDD datasets is represented in Table VII.

During the automated model learning process outlined in Table VII, the two best-performing models are selected as candidate models, and their hyperparameters are tuned using PSO to obtain the ultimate optimal learning model. As demonstrated in Table VII, the two most effective ML algorithms for both datasets are RF and LightGBM; hence, their hyperparameters are optimized. The search space and optimal values for the hyperparameters of these two learning algorithms are provided in Table VII. Continuous hyperparameters are assigned a search range, while categorical hyperparameters are given all potential values/options.

Table VII demonstrates the effectiveness of the proposed AutoML pipeline by presenting three sets of results. The first set of results displays the performance of ML algorithms after applying the proposed AutoDP and AutoFE procedures, emphasizing the impact of data quality enhancement through AutoML. Finally, the third set exhibits the performance of the complete AutoML pipeline, which includes AutoDP, AutoFE, automated model selection of the top-2 ML algorithms, and HPO procedures.

The AutoML pipeline begins with the execution of AutoDP and AutoFE, followed by the automated selection of the two best-performing learning models based on their F1-scores in the second set of results. Subsequently, the hyperparameters of the chosen models are optimized to obtain an ultimate optimal model with the highest F1-score, as illustrated in the third set of results. The top-performing configurations in each experimental set are highlighted in bold in Table VII.

The left part of Table VII summarizes the results of the proposed AutoML pipeline on the CICIDS2017 dataset. For the original ML models without any AutoML procedures, four models (KNN, MLP, RF, and LightGBM) exhibit large performance differences. The original RF and LightGBM models
C. Use Case 2: AML Attack and Defense

1) Experimental Setup: The second case study implements a cyber-defense exercise against three common AML attacks discussed in Section V-D2 (i.e., DTA [79], FGSM [80], and BIM [81]). These three AML attacks are launched in separate experiments to generate adversarial samples. These adversarial samples are intended to disrupt the input data used by the AutoML-based IDS obtained in the first case study. The purpose of these experiments is to demonstrate the potential damage caused by AML attacks on ML models within ZTNs/6G networks.

Subsequently, the experiments progress to the next stage where a fundamental AML defense mechanism, namely adversarial sample detection and filtering [13], is deployed against each type of AML attack. This illustrates the necessity and effectiveness of this fundamental defense mechanism in mitigating AML attacks and maintaining the reliability of the AutoML-based IDS within ZTNs/6G networks.

The specific procedures of the proposed cyber-defense exercise are as follows:

i) Evaluate the optimized IDS model obtained from the first case study’s AutoML procedure, LightGBM with the optimal hyperparameters outlined in Table VII on the original data set. This serves as the baseline model without any AML attacks.

ii) Generate adversarial samples using one of the AML attacks (i.e., DTA, FGSM, or BIM) and combine these samples with the original training set. This simulates a disrupted input dataset under the AML attack.

iii) Assess the IDS model on the adversarial samples generated in Procedure [13]. This demonstrates the performance of a model under an AML attack.

iv) Develop an adversarial sample detection model using the same LightGBM model. This model is trained on an integrated dataset of the original training set and adversarial samples to distinguish between them and identify adversarial samples.

v) Remove the detected adversarial samples from the training set to create a sanitized dataset.

vi) Re-train the IDS model on the sanitized training set and evaluate its performance on the test set, representing the effectiveness of the chosen defense model: adversarial sample detection and filtering.

In this case study, the optimized LightGBM models fulfill dual roles as both the models are susceptible to compromise under AML attacks and the defense models for detecting adversarial samples. However, their expected deployment environments differ. The defense models are expected to be deployed in a highly secure central server, while the compromised IDS models are presumed to be deployed in low-cost edge devices with generally inadequate security mechanisms.

2) Experimental Results and Analysis: The performance of the cyber-defense exercise against AML attacks was evaluated by comparing the accuracy, precision, recall, F1-score, and learning time for each ML model. After applying HPO to the two best-performing models, RF and LightGBM, the optimized LightGBM model attains the highest F1-score of 99.823%.

Similarly, the right part of Table VII reveals that RF and LightGBM models outperform the other two compared ML models on the 5G-NIDD dataset. Implementing AutoDP and AutoFE results in slight F1-score improvements and reduced learning time for each ML model. After applying HPO to the two best-performing models, RF and LightGBM, the optimized LightGBM model attains the highest F1-score of 99.877% and is chosen as the final optimal model for the 5G-NIDD dataset.

In conclusion, the AutoML pipeline generates a superior IDS model with F1-score improvements of 0.134% and 0.067%, and learning time reductions of 50% and 12.5% compared to the best-performing learning model without AutoML on the CICIDS2017 and 5G-NIDD datasets, respectively. This highlights the effectiveness of the AutoML pipeline in improving the quality of the IDS by effectively automating the ML process.
datasets. For the original IDS model without AML attacks, the model demonstrates a high degree of performance across all metrics: accuracy, precision, recall, and F1-score. Specifically, the F1-scores of the pristine IDS model stand at 99.556% and 99.899% on the CICIDS2017 and 5G-NIDD datasets, respectively.

However, the performance of the IDS model significantly decreases under the influence of DTA, FGSM, and BIM attacks on both datasets. The F1-scores of the IDS fall to a range between 0.134% and 69.658%, reflecting a decline of at least 30% across both datasets. As shown in Table VIII among the three AML attacks, DTA causes the most damage to the CICIDS2017 dataset, while BIM proves to be the most effective AML attack against the 5G-NIDD datasets. In addition, BIM attacks cause a significant increase in the model learning time for both datasets.

In contrast, the implementation of the adversarial sample detection and filtering mechanism results in a remarkable performance improvement for the IDS model. The accuracy, precision, recall, and F1-scores exhibit significant enhancements, closely resembling those of the original IDS model. On the CICIDS2017 dataset, the F1-scores of the restored IDS models range from 99.423% to 99.512%, whereas on the 5G-NIDD dataset, they range from 99.866% to 99.899%.

The performance of the AML detection model, as depicted in Table VIII, demonstrates its effectiveness in identifying and differentiating adversarial samples from the original, legitimate samples. For each form of AML attack, high accuracy, precision, recall, and F1-scores are recorded, ranging from 99.451% to 99.937%.

In summary, the destructiveness of AML attacks on ML models within ZTNs/6G networks can be seen in the following:

i) **Reduced Accuracy and Effectiveness**: AML attacks can significantly degrade the accuracy of ML-based IDSs, rendering them ineffective at detecting and mitigating security threats. This poses a threat to network resources and infrastructure.

ii) **Increased Complexity**: AML attacks add an additional layer of complexity to the already complex security landscape of ZTNs/6G networks, necessitating the development of defense mechanisms to defend against these sophisticated threats.

iii) **Erosion of Trust**: Successful AML attacks can erode confidence in ML models and their capacity to provide reliable security solutions for ZTNs/6G networks, resulting in decreased confidence in the adoption of these technologies.

Additionally, the adversarial sample detection and filtering method utilized in the case study can effectively distinguish adversarial samples generated by AML attacks and remove them. This results in a clean training set capable of effectively restoring the performance of ML models.

### VII. OPEN CHALLENGES AND RESEARCH DIRECTIONS

Despite the existence of various network security solutions, effectively deploying them within the context of ZTNs/6G networks poses several unresolved challenges. In this Section, the open challenges and research directions concerning ZTN security mechanisms are discussed. These challenges and directions are comprehensively summarized in Table IX.

#### A. General Cybersecurity Mechanism Development Challenges

1) **Rapid Evolution of Cyber-Attacks**: One of the major challenges of general cybersecurity is the rapidly evolving nature of cyber threats. Cybercriminals are constantly developing new methods to bypass existing security measures, and organizations must be able to adapt to these changes to maintain their security. Moreover, attackers can leverage AI/ML techniques to prioritize vulnerabilities for massive...
network attacks. For instance, AI-based botnets can identify zero-day vulnerabilities in IoT devices and exploit them to initiate DDoS attacks against 5G RAN resources [13]. This necessitates organizations to invest in the development and deployment of new cybersecurity technologies and to update existing security measures on a regular basis.

2) **Network Complexity**: Another challenge of cybersecurity is the complexity of 5G and 6G networks [74]. As the number of connected devices increases and cloud computing becomes prevalent, detecting and responding to cybersecurity threats in complex networks becomes increasingly difficult. Numerous systems and devices require security measures, and identifying the most vulnerable components can be challenging.

3) **Cybersecurity System Cost**: The financial burden of cybersecurity poses a significant challenge, particularly for small enterprises. The cost of implementing effective security measures must be measured against the cost of a security violation [93]. Additionally, quantifying the cost of network security is challenging as it encompasses not only the expenses related to implementing security measures but also potential costs arising from data breaches, delays, and reputational damage.

4) **Cross-Layer Network Security**: Traditional ML-based network management services focus on single protocol stack layers, limiting large-scale intelligence potential. Cross-layer cooperation, however, enables the development of more flexible and efficient solutions in heterogeneous networks [14]. In the context of cross-layer network security, it is essential to examine the interdependencies between different layers of the protocol stack. For instance, the RAN can optimize wireless channel operation by dynamically adapting both the link and physical layers accordingly. By leveraging cross-layer intelligence, it becomes possible to identify and address security challenges that span multiple layers, leading to more comprehensive and robust security mechanisms.

Cross-layer network security solutions can also facilitate more effective detection and response to cybersecurity threats, as the integration of information from different layers can provide a more accurate and complete understanding of the network’s security state. This holistic approach can lead to the development of more advanced ML-based security models that can adapt to evolving threats, improving the overall resilience of the network.

### B. AML Defense Model Development Challenges

1) **Lack of Autonomous Solutions**: Existing AML security models typically require human analysis or supervision to comprehend attack patterns and design effective countermeasures. However, autonomous cybersecurity solutions are required to facilitate network automation for ZTNs/6G networks [8]. AutoML techniques discussed in Section V-C are promising solutions for interacting with networking environments and developing security models automatically. However, current AutoML techniques are proposed for general cybersecurity issues, and additional research is necessary to address AI/ML model security challenges.

2) **Feasible Cyber-Defense Exercises**: Cyber-defense exercises that simulate cyber-attacks in networks are often required...
to develop and evaluate AML countermeasures and security mechanisms, but launching attacks will disrupt the normal operation of physical networks [94]. Digital Twin (DT) [95] is a promising technology for improving traditional cyber-defense exercises. The primary objective of DT models is to generate virtual copies of a physical system or network to simulate its behavior and real-time usage. DT models enable cyber-defense exercises and security model evaluations in cybersecurity applications without interfering with the normal operation of networks. In DT-assisted networks, AML attacks and defense operations can be simulated by launching multiple types of AML attacks and then evaluating the effectiveness of protection mechanisms. Once their performance is evaluated, the defensive models will be transmitted back to the actual network from the virtual networks.

3) Model Privacy Issues: The implementation of certain AML countermeasures for 5G/6G networks raises concerns regarding model privacy, as it necessitates the sharing of data and models from multiple network devices to develop comprehensive security measures [78]. Developing effective countermeasures for specific attacks usually requires an exhaustive and detailed comprehension of AML attacks and affected ML models. However, data or model leakage may occur during the development of AML security models. Federated learning [96] and blockchain [97] techniques are potential solutions to address the privacy issues of AML security model development, but their integration with AML security approaches requires additional research.

4) Interpretability and Explainability: Explainable Artificial Intelligence (XAI) is a crucial aspect of developing and deploying AML protection models in B5G/6G networks [98]. Even though ML models have the potential to achieve high levels of accuracy and performance, their decision-making processes are often viewed as black boxes, which makes it challenging for users to understand the reasoning behind their conclusions. The limited interpretability of AML models can impede the development of effective countermeasures, as security analysts face challenges in identifying potentially exploitable vulnerabilities within the model. Incorporating XAI techniques into AML protection models can provide a deeper understanding of the decision-making process, enabling the identification of vulnerabilities and the development of more robust defenses against adversarial attacks. However, achieving the optimal balance between model performance, interpretability, and security remains a formidable task that necessitates further research.

C. Privacy Preservation Technology Development Challenges

Privacy protection is a critical challenge in 5G/6G networks, primarily due to the collection and sharing of exponentially increasing network data. Blockchain, federated learning/distributed learning, quantum homomorphic encryption, and differential privacy are a few potential privacy-enhancing technologies that will have an influence on 6G [18]. To achieve a practical deployment of 6G, however, several critical shortcomings and vulnerabilities must be addressed.

1) Financial input: Implementing robust privacy protection measures involves meticulous consideration of multiple factors. One critical factor is the potential financial burden on companies, as ensuring data privacy necessitates substantial investments. This expense not only involves software alterations but also substantial investments in comprehensive security technologies for data encryption and anonymization. In the age of data proliferation, as exemplified by 6G, handling massive data processing might demand high-performance computing resources, potentially escalating costs beyond the company's financial capabilities.

2) Limited Resources of IoT devices: Protecting privacy in large IoT devices with limited resources is a primary challenge. Rarely are low-cost IoT sensors and wearable devices provided with robust authentication and security procedures. The absence of sophisticated security could undermine the defenses against massive data collection. In 6G, when smart objects and people are intended to be interconnected, network security concerns will grow substantially. Addressing this challenge requires innovative approaches such as TinyML [99], which focuses on developing lightweight ML models tailored for low-configuration devices.

3) Community awareness: Community awareness of privacy rights and well-organized norms are essential for accelerating the development of technologies that enhance privacy. Consumers may be reticent to disclose their confidential information to utilize 6G network application services due to privacy concerns. Without updated policies and laws, it could take years or even decades to resolve all problems. Organizations are more likely to prioritize data privacy when faced with stringent legal penalties and stringent end-user requirements.

D. Challenges of AutoML Techniques

During the past decade, AutoML has made substantial progress in automating model construction and development, especially for supervised learning tasks. However, AutoML faces several challenges that need to be addressed before it can be widely applied to real-world ZTN/6G network applications [100]:

1) Automated Data Pre-Processing and Feature Engineering: Although there are many existing AutoML solutions, the vast majority of them concentrate on automated model selection and HPO, with less emphasis placed on automated data pre-processing and feature engineering [100]. Yet, these two aspects are vital components of the AutoML pipeline and directly influence the performance of systems, particularly in the context of B5G/6G networks that require large-scale data analytics. Feature engineering is especially challenging to automate and generalize, as it heavily depends on the specific task and dataset [56]. Implementing effective feature engineering often necessitates specialized domain knowledge or considerable effort. As such, automated feature engineering, despite its complexity, is an imperative research topic that warrants further investigation.

2) Large Scale AutoML: AutoML’s application to large-scale data remains unresolved. Since AutoML pipelines often
need a large number of model trainings to determine the optimal end learner, most AutoML solutions are created on small datasets, with just a handful capable of large-scale data learning. For example, due to the huge size of the ImageNet dataset, research on AutoML solutions for the ImageNet challenge is still limited [44].

3) Trustworthy AI: Trustworthiness of AI/ML techniques refers to the extent to which an AI system is reliable, safe, secure, and ethical [101]. The development of trustworthy AI systems is crucial for ensuring that AI can be used for the benefit of ZTNs/6G networks rather than causing harm. Trustworthy AI is particularly important for AutoML models, as they often lack direct human supervision. Key aspects of AI trustworthiness include robustness, reproducibility, explainability, privacy, accountability, and fairness [102].

i) Robustness: It is a key aspect of AI systems, which refers to their ability to handle execution errors, incorrect inputs, and unseen data. A robust AI system should exhibit resilience at different levels of data, algorithms, and systems.

ii) Reproducibility: It indicates the ability of AI/ML models to be reproduced. It is crucial for validating and assessing AI/ML research, enabling the identification and mitigation of potential risks in AI systems. The AI research community increasingly views reproducibility as a requirement for publishing research.

iii) Explainability: It is essential for building trust in AI/ML technology and entails understanding the decision-making process of AI/ML models [103]. Research on AI/ML explainability has been conducted in two aspects. The first aspect involves the development of fully or partially explainable ML models, such as linear regression and decision tree-based models. The second aspect focuses on analyzing complex models like DL models by examining their input, intermediate results, and output.

iv) Privacy Protection: It refers to safeguarding data that can directly or indirectly identify individuals or households. Privacy protection spans the entire lifecycle of AI/ML systems, covering data collection & pre-processing, model training, and deployment.

v) Accountability: It is essential for ensuring that AI systems adhere to trustworthiness requirements. Accountability incorporates the entire lifecycle of an AI system and requires stakeholders to justify their design, implementation, and operation in accordance with human values.

vi) AI Fairness: It involves mitigating various types of bias, such as data bias, model bias, and procedural bias [104]. These biases frequently lead to the unjust treatment of various groups based on their protected information, such as gender, race, and ethnicity. To prevent the perpetuation or exacerbation of social bias, it is vital to resolve impartiality in AI systems.

In conclusion, establishing trust in AI/ML and AutoML technologies is a critical step in their successful application to ZTNs/6G networks.

VIII. Conclusion

The rapid emergence of sixth-generation wireless networks (6G) and the increasing demand for network automation have led to the development of Zero-Touch Networks (ZTNs), where Artificial Intelligence (AI) and Machine Learning (ML) play a critical role. However, successful deployment of ZTNs requires resolving numerous security challenges during their implementation. In this survey paper, we have conducted a comprehensive review of the security issues and vulnerabilities associated with ZTNs and explored potential solutions, with a specific focus on Automated ML (AutoML) technologies and Adversarial ML (AML) attacks and defense mechanisms. Two case studies are presented in this paper. The first case study focuses on the development of autonomous frameworks to address security issues in ZTNs, demonstrating the effectiveness of AutoML technologies in securing ZTNs. The second case study on the cyber-defense exercises of AML attacks has been discussed to illustrate the devastating impact of AML attacks, along with potential defense mechanisms. Moreover, open challenges and future research directions for ZTN security research have been discussed, highlighting the need for ongoing research and innovation in this field.

REFERENCES


A. L’Heureux, K. Grodinger, H. F. Elyamany, and M. A. M. Capretz, "Machine Learning with Big Data: Challenges and Approaches," Springer Interna-


Abdallah Shami  is currently a Professor in the Electrical and Computer Engineering Department and the Acting Associate Dean (Research) of the Faculty of Engineering, Western University, London, ON, Canada, where he is also the Director of the Optimized Computing and Communications Laboratory. Dr. Shami has chaired key symposia for the IEEE GLOBECOM, IEEE International Conference on Communications, and IEEE International Conference on Computing, Networking and Communications. He was the elected Chair for the IEEE Communications Society Technical Committee on Communications Software and the IEEE London Ontario Section Chair. He is currently an Associate Editor of the IEEE Transactions on Information Forensics and Security, IEEE Internet of Things Journal, and IEEE Communications Surveys and Tutorials journals.

Sami Muhaidat  received the Ph.D. degree in Electrical and Computer Engineering from the University of Waterloo, Waterloo, Ontario, in 2006. From 2007 to 2008, he was an NSERC Postdoctoral Fellow in the Department of Electrical and Computer Engineering, University of Toronto, Canada. From 2008-2012, he was an Assistant Professor in the School of Engineering Science, Simon Fraser University, BC, Canada. He is currently a Professor at Khalifa University and an Adjunct Professor with Carleton University, Ontario, Canada. Sami’s research interests focus on advanced digital signal processing techniques for wireless communications, intelligent surfaces, MIMO, optical communications, massive multiple access techniques, backscatter communications, and machine learning for communications. He is currently an Area Editor of the IEEE Transactions on Communications, a Guest Editor of the IEEE Network “Native Artificial Intelligence in Integrated Terrestrial and Non-Terrestrial Networks in 6G” special issue, and a Guest Editor of the IEEE OJVT “Recent Advances in Security and Privacy for 6G Networks” special issue. He served as a Senior Editor and Editor of the IEEE Communications Letters, an Editor of the IEEE Transactions on Communications, and an Associate Editor of the IEEE Transactions on Vehicular Technology.