Security And Dependability Challenges In Virtualized Networking And Computing Environments

Akshara PB[1], Prof.Hasna M[2]
[1, 2] Dept of Computer Science and Engineering

Abstract- *The Fifth Generation (5G) of mobile networks introduces transformative services with strict requirements for performance, security, and dependability. Multi-access Edge Computing (MEC), a cornerstone of 5G, enhances network efficiency by reducing latency, enabling real-time local awareness, supporting cloud offloading, and mitigating traffic congestion. These advancements are critical for mission-critical applications but come with security and reliability challenges that are of- ten underexplored alongside performance. This survey paper addresses this gap, presenting a comprehensive analysis of 5G MEC's security, dependability, and performance, along with state-of-the-art solutions and challenges in these domains. Inter- net of Things (IoT) security has garnered significant attention due to its vulnerability to attacks such as Denial of Service (DoS) and data breaches. A novel machine learning (ML)-based security framework leveraging Software Defined Networking (SDN) and Network Function Virtualization (NFV) is proposed to counter these threats. This framework employs AI-driven monitoring and anomaly-based intrusion detection, demonstrating high accuracy (99.715G also enables diverse services through virtualization, softwarization, and network slicing, fostering the rise of Mobile Virtual Network Operators (MVNOs). While these technologies enhance flexibility, they also introduce complex security implications. This paper reviews security challenges and potential solutions for MVNOs, emphasizing the critical role of robust virtualization techniques, such as microkernel-based environments, which enhance resource utilization and security. In the domain of Smart Grids (SG), integrating power networks with information technologies introduces vulnerabilities due to increased automation and connectivity. The application of ML in SG enhances attack detection and threat analysis but also exposes systems to adversarial ML attacks. This survey examines the security and privacy challenges of SG, presenting taxonomies and novel findings to address these issues.*

I. INTRODUCTION

The Fifth Generation (5G) of mobile networks represents a transformative era in wireless communication, poised to redefine the digital landscape as we know it. With its unpar- alleled capabilities, 5G is not just an evolution of existing networks but a revolution that integrates cutting-edge tech- nologies to deliver extraordinary connectivity. Its promise of ultra-high speeds, massive capacity, and ultra-reliable low- latency communication (URLLC) opens doors to countless possibilities across industries and daily life. Designed with three primary use cases in mind—enhanced Mobile Broadband (eMBB), massive Machine-Type Communication (mMTC), and URLLC—5G is tailored to meet diverse demands, from immersive entertainment experiences to mission-critical appli- cations.

eMBB addresses the need for blistering internet speeds, deliv- ering seamless 4K/8K streaming, real-time cloud gaming, and augmented reality (AR) applications, while also empowering remote work and collaboration. mMTC focuses on connecting billions of IoT devices, enabling smart homes, intelligent infrastructure, and advanced agricultural practices. URLLC, on the other hand, ensures the reliability and responsiveness required for groundbreaking applications like autonomous vehicles, remote surgeries, and industrial robotics. The con- vergence of these capabilities marks a significant leap towards a fully connected and automated society.

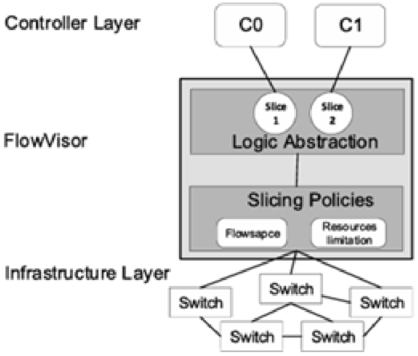

Fig. 1. Flow Visor Architecture [1]

The Fig 1 [1] illustrates a simplified SDN architecture with network slicing capabilities. The Flow Visor, the central con- trol point, collects network information and makes decisions about how to configure the network. It then sends instructions to individual controllers, which in turn configure switches in the infrastructure layer. Network slicing allows for the creation of isolated virtual networks, each with its own set of resources and policies. This enables flexible and customized network configurations, supporting a wide range of network services and applications [1].

II. SECURITY CHALLENGES

A. Vulnerabilities in Virtualization Layers

Virtualization is a cornerstone technology in modern computing systems, enabling the abstraction of physical hardware into multiple virtual environments. While it offers significant advantages, such as resource efficiency and scalability, it also introduces unique security vulnerabilities within its virtualization layers. The (In)Security of Virtualization in Software-Defined Networks [2] explores these vulnerabilities in detail, particularly in the context of Software-Defined Networks (SDNs).

Key Vulnerabilities in Virtualization Layers:

1) *Hypervisor vulnerabilities:* It pose a significant risk, as attackers can exploit them to escalate privileges and gain unauthorized control over all hosted virtual machines. Additionally, side-channel attacks leverage shared resources, such as CPU caches, to infer sensitive data across virtual machine boundaries.
2) *VM Escape:* This occurs when a malicious VM breaches the isolation barrier to access the underlying hypervisor or other VMs on the same host. Attackers exploit bugs in the hypervisor or virtual device emulation. Examples include exploiting memory management errors or buffer overflows.
3) *Insecure Inter-VM Communication:*Virtualized environments often rely on inter-VM communication for efficiency, but insecure configurations can lead to data leakage or man-in-the-middle attacks [3]. Poorly configured virtual networks or APIs can expose sensitive data.
4) *ResourceContention and Denial of Service (DoS):*Shared resource pools in virtualized environments can lead to contention. Malicious VMs can consume excessive resources, causing performance degradation or denial of service to other VMs.
5) *Virtual Network Vulnerabilities:* Virtualized SDNs of- ten rely on virtual switches and controllers for traf- fic management, making them susceptible to various security threats. Attackers can exploit weaknesses in the SDN controller, potentially compromising the entire virtualized network. Additionally, misconfigurations or vulnerabilities in virtual switches can lead to data plane attacks, allowing adversaries to inject malicious packets or eavesdrop on network traffic.
6) *Configuration Errors:*Misconfigurations during setup, such as insecure APIs, weak authentication mechanisms, or inadequate access controls, increase the attack surface. Real-world incidents highlight virtualization vulnerabilities, such as the exploitation of Open vSwitch flaws leading to data plane compromises and attackers leveraging hypervisor weaknesses in Xen and VMware to escape VM isolation. To mitigate these risks, Paper 6 proposes several security measures, including hypervisor hardening through regular updates and minimizing the attack surface by disabling unnecessary features. Enhanced isolation mechanisms, such as Intel VT-x and AMD-V, provide hardware-assisted security to maintain VM boundaries. Secure communication protocols, enforced through encryption and authentication, prevent unauthorized access and data breaches. Additionally, resource monitoring helps detect and mitigate abuse, reducing the risk of performance degradation or denial-of-service attacks. For Software-Defined Networks (SDNs), securing controllers and virtual switches with robust access controls and continuous monitoring is essential to maintaining network integrity. These combined strategies offer a comprehensive defense against virtualization layer vulnerabilities.

B. Machine Learning-Based Security Threats in IoT Systems

The rise of the Internet of Things (IoT) has introduced new opportunities and challenges, particularly in the realm of secu- rity. IoT systems, comprising a vast network of interconnected devices, are increasingly targeted by cyber-attacks due to their widespread use, diverse communication protocols, and often limited security mechanisms. One of the emerging threats in this space is the exploitation of machine learning (ML)-based security vulnerabilities, provides authors in [3].

Key ML-Based Security Threats in IoT Systems:

1) *Adversarial Attacks on ML Models:* In IoT environments, machine learning models are increasingly deployed for tasks such as anomaly detection, traffic prediction, and security monitoring. However, these models are vulnerable to adversarial attacks [3], where attackers introduce subtle perturbations to input data to cause misclassifications or errors in the predictions made by the ML model. For example, a malicious actor could manipulate sensor data to mislead an IoT device's security system into overlooking a potential intrusion. Example: Modifying temperature or motion sensor data to bypass intrusion detection algorithms in smart homes.

2) *Model poisoning:* It involves attacking the training process of a machine learning model by injecting malicious data into the dataset used to train the model. This can cause the model to learn incorrect patterns or behave in a way that benefits the attacker. In IoT systems, where data is continuously generated by a multitude of devices, attackers can corrupt the training data by introducing faulty or malicious data streams from compromised IoT devices [3].

   Example: A compromised smart device sending manipulated data that affects the training of a machine learning-based security model for the entire IoT network.

3) *Data Privacy and Confidentiality Risks:* Machine learning models in IoT systems often require access to large amounts of personal or sensitive data to operate effectively. Data inference attacks [3] pose a significant threat to data privacy, as attackers may use the ML model to infer private information about individuals or organizations. This is particularly concerning when IoT devices are embedded in sensitive environments, such as healthcare or home automation systems.

   Example: Extracting private information, such as health conditions or personal habits, from data used to train ML models in smart health devices or personal assistants [3].

4) *Denial of Service (DoS) attacks:* It aim to overwhelm an IoT system with excessive data or requests, rendering it unresponsive. In the case of ML-based security models, attackers can overload the system with malformed or noisy data, causing the model to fail or degrade in performance. In IoT systems that rely on real-time decision-making, such as autonomous vehicles or industrial control systems, this can have catastrophic consequences. Example: Sending excessive false alerts to a security model in an IoT network, causing it to malfunction or delay in detecting real security threats.

5) *Evasion Attacks on Intrusion Detection Systems (IDS):* Many IoT security systems rely on machine learning-based Intrusion Detection Systems (IDS) to detect anomalous or malicious behavior. However, attackers can evade detection [3] by crafting data that is specif- ically designed to bypass these systems, often using techniques like feature manipulation or data obfuscation. Example: An attacker modifying network traffic in a way that avoids detection by an IDS trained to recognize specific malicious patterns in data traffic.

6) *Unintentional Bias in ML Models:* Machine learning models, especially when trained on imbalanced or incomplete datasets, can inherit biases that compromise the security of IoT systems. In IoT environments, where devices from diverse sources with varied capabilities are deployed, biases in the security models can lead to vulnerabilities that attackers can exploit [3].

   Example: A model that incorrectly classifies legitimate traffic from certain IoT devices as malicious due to biases in the training dataset, leading to security breaches or system failures.

C. ***Security concerns in virtual mobile networks***

Virtual Mobile Networks (VMNs) represent a significant evolution in mobile telecommunications, where network infrastructure is abstracted and virtualized to support flexible, scalable, and cost-effective services. However, while VMNs provide several advantages, including the ability to efficiently manage resources and deploy new services, they also introduce a new set of security concerns, [4] by I. A. Jani et al. explores security issues in depth, providing insights into the vulnerabilities specific to VMNs.

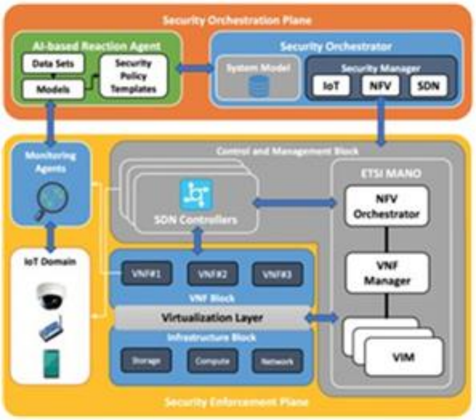

Fig. 2. Proposed Framework Main Overview [3]

Fig. 2 [3] describes the communication framework for IoT systems, highlighting the interaction between VNFs, PNFs, IoT devices, and end-users over legacy or SDN-based networks. It also clearly distinguishes between internal and external attacks and outlines mitigation strategies at the IoT, network, and cloud/MEC levels. Additionally, the discussion on security enforcement using SDN/NFV standards aligns well with industry specifications. However, the readability could improve with clearer sentence structures, especially when discussing attack origins and mitigation levels.

Key Security Concerns in Virtual Mobile Networks:

1) *Network Slicing Vulnerabilities:* One of the foundational concepts in VMNs is network slicing, where a physical network is divided into multiple virtual networks that can be customized for specific use cases. While network slicing provides flexibility, it also creates new attack surfaces. Isolation Failures: If proper isolation between slices is not maintained, an attacker in one slice can potentially access resources or services in another slice. Slice Management Compromise: Attackers could target

the management of virtualized slices to manipulate the configuration, leading to denial-of-service (DoS) attacks, resource misallocation, or data theft.

2) *Virtualized Infrastructure and Hypervisor Risks:* VMNs rely on virtualization technologies, and like other virtualized environments, they are susceptible to the same hypervisor vulnerabilities [4]. The hypervisor, which manages the virtualized network functions (VNFs) in VMNs, is a critical point of failure. VM Escape: Attackers could exploit hypervisor vulnerabilities to break out of a virtual machine (VM) and gain control over the entire infrastructure. Denial of Service: The hypervisor can be targeted with DoS attacks, which may disrupt the operation of multiple VNFs or the entire virtual network.
3) *Compromised Virtual Network Functions (VNFs):* VNFs are software-based network functions that replace traditional hardware-based appliances, such as routers, firewalls, and load balancers, in a VMN. If these functions are compromised, it can lead to significant disruptions in the network. Attackers may target VNFs through hijacking, manipulating their behavior to redirect traf- fic, perform eavesdropping, or inject malicious data. Additionally, VNF misconfiguration, whether accidental or intentional, can expose vulnerabilities, making the network prone to attacks such as man-in-the-middle (MITM) or unauthorized access.
4) *Data Privacy and Confidentiality Issues:* In VMNs, data is transmitted across virtualized environments that may be shared by multiple operators or tenants, raising significant concerns regarding data privacy and confidentiality [4]. Insecure data channels or weak isolation between virtual network functions can result in accidental or malicious data leakage, exposing sensitive user information. Additionally, without proper security measures, one tenant or virtual network may gain unauthorized access to another tenant’s data, violating privacy regulations and compromising confidentiality.
5) *Control Plane Attacks:* The control plane of a VMN, which is responsible for managing network traffic, configuring VNFs, and handling signaling, is a potential target for attackers. A compromised control plane can lead to full network compromise. Attackers could manipulate control plane messages to disrupt communication, spoof identities, or hijack control over VNFs. Exploiting the control plane can lead to malicious rerouting of data or creation of unauthorized network paths.
6) *Interoperability and Standards Compliance:* VMNs often rely on a diverse set of technologies, vendors, and standards, creating challenges in ensuring interoperability and consistent security across the system [4]. The lack of uniform security standards or incompatible secu- rity mechanisms across different vendors or virtualized components can result in gaps in protection. Additionally, integrating legacy systems with virtualized systems or third-party services may introduce vulnerabilities if proper security measures are not enforced.
7) *Vulnerability to Distributed Denial of Service (DDoS) Attacks:* VMNs are exposed to a higher risk of DDoS attacks, as virtualized infrastructure and shared resources make it easier for attackers to overwhelm specific network components. A successful DDoS attack can impact the availability of essential services, such as VoIP, video streaming, and online transactions. Attackers could leverage the virtualization layer to amplify DDoS attacks, targeting network resources or virtual func- tions.This amplification can result in cascading failures across multiple virtual machines, significantly disrupting network operations and service continuity.

D. Virtualization Vulnerabilities in SDN and Embedded Systems

Virtualization plays a pivotal role in Software-Defined Net- works (SDN) and embedded systems, offering flexibility, scal- ability, and cost-effective management of resources. However, the introduction of virtualization also introduces a range of vulnerabilities that can compromise the security, dependability, and performance of both SDNs and embedded systems. T [5]hese vulnerabilities stem from the unique architecture of virtualization, where physical resources are abstracted and shared between multiple virtual entities. In the context of SDN, which centralizes network control and management, and embedded systems, which are often deployed in critical and resource-constrained environments, the risks posed by virtualization become even more pronounced. In SDNs, the centralization of control in a software-based controller makes it a high-value target for attackers. If the hypervisor or con- troller is compromised, attackers can gain full control over the network’s data plane, enabling them to reroute traffic, launch denial-of-service (DoS) attacks, or introduce malicious behav- ior across the entire network. The hypervisor vulnerabilities in SDN systems, such as exploitation of flaws in the virtualization layer or the control plane, can allow attackers to gain unauthorized access to virtualized network functions (VNFs) or break isolation between network slices. This can lead to cross-tenant attacks and data leakage, exposing sensitive information and disrupting the entire virtual network infrastructure.

In embedded systems, the risks of virtualization are simi- larly concerning. These systems, which are often used in critical applications like industrial automation, healthcare, and autonomous vehicles, rely on real-time processing and specialized hardware. The use of virtualization in these systems can introduce performance bottlenecks and increase the attack surface due to shared resources such as memory, storage, and processing power. A compromised virtual machine can affect the stability and operation of the entire embedded system, as attackers may exploit vulnerabilities in the hypervisor or in virtualized hardware devices. Virtual machine escape [5], where an attacker breaks out of a compromised VM to access the underlying host system, is a particularly dangerous vulnerability that can disrupt embedded systems' real-time capabilities or cause system-wide failures.

Moreover, both SDN and embedded systems are increasingly dependent on external software components and third-party services, which increases the complexity of securing these en- vironments. Weaknesses in these components, whether in the form of insecure APIs, misconfigurations [6], or lack of regular updates, further expose the system to potential breaches. To mitigate these vulnerabilities, organizations must adopt robust security practices, such as secure virtualization techniques, segmentation, and real-time monitoring to detect abnormal behavior. Regular patching of both hypervisors and virtualized components, combined with a layered approach to security, can help reduce the risks posed by virtualization vulnerabilities in SDN and embedded systems [6]. In conclusion, while virtualization provides significant benefits to SDN and embed- ded systems, it also introduces critical security vulnerabilities that must be addressed through proactive security measures. Ensuring strong isolation, continuous monitoring, and secure configurations will be key in protecting these environments from potential attacks and system failures.

E. Threats in Smart Grid Security

”Smart Grid Security and Privacy: From Conventional to Machine Learning Issues (Threats and Countermeasures)”, delves into the multifaceted security challenges faced by smart grid systems. As smart grids integrate advanced communication technologies, sensors, and automated control mechanisms, they become increasingly susceptible to both cybersecurity and physical threats [5]. It highlights various threats that could undermine the security, privacy, and functionality of the grid, alongside evolving countermeasures leveraging machine learning and advanced algorithms to address these issues. The following outlines the primary threats identified in the paper.

1) Cybersecurity Threats: The expansion of smart grid technologies introduces vulnerabilities that can be ex- ploited by cyber attackers, targeting both the cyber infrastructure and the communication networks integral to smart grid operations.

Data Breaches and Privacy Issues: Smart grids collect sensitive data through smart meters, sensors, and other devices that monitor energy consumption patterns. If compromised, this data could be used maliciously, revealing private consumer behavior or even enabling identity theft. Example: Attackers gaining unauthorized access to smart meter data and using it to infer personal information or manipulate billing

Denial-of-Service (DoS) Attacks: A DoS attack can disrupt communication channels, preventing data flow between grid components, affecting real-time monitoring, control, and management. Distributed Denial-of-Service (DDoS) attacks [5], in particular, can target critical network resources, causing significant operational delays or service outages. Example: A DDoS attack on the control systems, which causes grid operators to lose visibility into critical infrastructure, potentially leading to mismanagement of grid resources. Man-in-the-Middle (MITM) Attacks: MITM attacks are particularly dangerous in smart grids, where attackers intercept and manipulate data as it travels between smart meters, sensors, and control systems. By altering readings or commands, they can disrupt the entire energy management system [7]. Example: An attacker intercepting data from a smart meter to change consumption readings, leading to incorrect billing or the activation of faulty control signals.

Ransomware and Malware: Malware and ransomware attacks can lock grid operators out of critical systems or spread throughout the network to take control of grid infrastructure. Such attacks can result in the temporary or permanent loss of grid control or the disruption of vital energy supply. Example: Ransomware locking operators out of their control systems, demanding payment to restore access to vital grid operations, such as energy distribution management.

2) Physical Security Threats: While many of the threats to smart grids are cyber-based, physical security remains a crucial concern. The smart grid relies on physical components like substations [7], transformers, power lines, and communication hubs, which are vulnerable to sabotage or attacks.

Sabotage and Physical Attacks on Infrastructure: Attackers can target physical infrastructure, such as power substations, transformers, and communication nodes, to cause damage or interrupt grid operations. The loss of key equipment can result in widespread outages and significant recovery costs. Example: A coordinated attack on a substation that disables key components, causing widespread power loss.

Tampering with Smart Meters and Devices: Smart meters, which monitor energy consumption, are an essential part of the smart grid infrastructure. If tampered with, they can provide incorrect data to the grid, leading to billing errors or disrupting grid management functions. Example: Attackers physically tampering with smart meters to either lower energy consumption readings or disrupt their communication with central systems.

3) Insider Threats: Given the complexity of smart grid systems and their reliance on various stakeholders, insider threats remain a significant concern. Individuals with access to critical systems, maliciously or negligently, can cause significant harm [7].

Malicious Insiders: Employees or contractors with privileged access to the smart grid infrastructure can intentionally exploit vulnerabilities for financial gain, political reasons, or personal reasons. These insiders can compromise systems or data, disrupting grid functions. Example: A disgruntled employee intentionally altering grid control settings, leading to resource mismanagement or disruption of services.

Negligence and Mismanagement: Even without malicious intent, improper handling of sensitive information or failure to maintain secure protocols can lead to breaches or security lapses. This includes using weak passwords, failing to update software, or inadvertently granting excessive access rights. Example: A technician inadvertently leaving a system endpoint open for attackers to exploit or failing to properly update security patches on network devices.

4) Supply Chain Vulnerabilities: The smart grid is built using components and technologies from multiplevendors, which introduces potential vulnerabilities through the supply chain. Attackers may target weaknesses in third-party products or services to compromise the grid’s security.

Compromised Hardware and Software: The introduction of compromised or poorly tested components into the smart grid system can lead to vulnerabilities that attackers can exploit. This includes the integration of unsecure hardware, firmware, or software that may contain hidden backdoors or weaknesses. Example: A supplier providing insecure smart meters or communication devices, which contain vulnerabilities that allow attackers to remotely access and control grid components.

Third-Party Service Providers: Smart grid systems often depend on third-party service providers for software updates, maintenance, and data analysis. If these service providers are not properly vetted or secured, they can become a potential entry point for attackers. Example: A service provider delivering malicious software updates that compromise grid security when integrated into the system.

5) Communication Network Vulnerabilities: The communication networks that connect the diverse elements of the smart grid are susceptible to various attacks due to the reliance on public and private communication infrastructure.

Weaknesses in Communication Protocols: The protocols used for data transfer between grid components, such as the Advanced Metering Infrastructure (AMI), may be vulnerable to interception, manipulation, or spoofing [5]. Without secure encryption, these communications can be hijacked or altered to mislead operators. Example: Attackers exploiting weaknesses in communication protocols to inject false data into the grid control system, affecting the energy supply chain.

Lack of Standardization: The integration of various technologies and devices from different manufacturers often leads to a lack of consistent security measures. Disparities in the security protocols between devices and vendors can create gaps in the overall security architecture of the network. Example: An attacker exploits protocol differences between devices or vendors to launch an attack on the communication infrastructure.

6) Privacy and Data Protection Issues: The extensive data collection by smart grids, including consumer behavior, energy consumption patterns, and operational statuses, raises significant privacy concerns. Protecting this data from unauthorized access and misuse is critical. Unauthorized Access to Consumer Data: As smart grids collect detailed information about energy usage, unau- thorized access to this data can lead to privacy violations. Data could be sold, misused, or exposed to external parties without consumer consent [7]. Example: Hackers gaining access to detailed consumer usage data and selling it to third parties for targeted advertising or identity theft.

Data Integrity and Accuracy: Ensuring the accuracy and integrity of the data being transmitted and processed within the smart grid is essential. Attackers may alter energy usage data to manipulate pricing, billing, or system operations. Example: Malicious actors altering energy usage data to generate fraudulent billing records or manipulate market pricing.

To address these diverse threats, several countermeasures have been proposed. Implementing advanced encryption protocols for all data transmissions within the smart grid helps protect sensitive information from interception or tampering. Deploying Intrusion Detection and

Prevention Systems (IDPS) enables real-time monitoring to quickly detect and respond to suspicious activities or potential breaches. Additionally, leveraging machine learning and artificial intelligence for threat detection al- lows systems to predict, identify, and mitigate emerging threats by recognizing patterns of anomalous behavior. Strengthening physical security measures at critical grid infrastructure sites is essential to prevent tampering and sabotage. Furthermore, conducting regular software and hardware audits ensures the integrity and security of all components, particularly those sourced from third-party vendors.

III. DEPENDABILITY ISSUES

A. Dependability of Hardware Performance Counters in Embedded Systems

Resource Constraints pose a significant challenge in en- suring the dependability of Multi-Access Edge Computing (MEC) and IoT systems. Unlike centralized cloud environments, edge nodes often operate with limited computational power, storage capacity, and bandwidth. These limitations can lead to performance bottlenecks, especially when handling data-intensive tasks such as real-time analytics and machine learning applications. Furthermore, the constrained storage resources of edge devices may not support the large vol- umes of data generated by IoT systems, increasing the risk of data loss. Bandwidth limitations exacerbate the issue [2] by introducing delays in data transmission, particularly in scenarios requiring low-latency responses. Overcoming these constraints requires optimized resource allocation techniques, edge caching strategies, and scalable infrastructure design to balance load distribution and prevent resource exhaustion.

Fault Tolerance is another critical aspect that impacts the dependability of MEC and IoT systems. Edge nodes are prone to hardware failures, software crashes, and sudden discon- nections, leading to potential service interruptions. Unlike centralized systems, which often have redundant resources, edge environments operate with minimal backup mechanisms, making recovery from failures more complex. The dynamic nature of MEC [2] networks further adds to the challenge, as unexpected events can cause disruptions in service avail- ability. To address these concerns, redundancy mechanisms such as data replication, checkpointing, and rollback recovery techniques are being developed to enhance fault tolerance and ensure seamless operation during failures.

Heterogeneous Infrastructure also creates dependability chal- lenges in MEC and IoT systems. The integration of diverse hardware platforms, including IoT sensors, gateways, and edge servers, results in variations in computational capa- bilities, protocols, and software architectures. This diversity complicates interoperability and makes it difficult to maintain consistent performance across distributed environments [3]. Managing these heterogeneous resources requires advanced or- chestration frameworks and standardized protocols to stream- line communication and resource sharing. Technologies like containerization and virtualization are being widely adopted to improve compatibility and simplify deployment, enabling efficient management of distributed edge nodes.

Mobility Management presents additional dependability chal- lenges due to the dynamic movement of users and devices within MEC environments. Mobile devices frequently switch between edge nodes, resulting in service disruptions and performance degradation. This issue is particularly critical for latency-sensitive applications such as autonomous ve- hicles, augmented reality, and industrial automation, where uninterrupted connectivity is essential. Moreover, dynamic allocation of resources to meet mobility demands often leads to resource shortages, which affect the reliability of services [3]. To address these challenges, predictive algorithms are being developed to anticipate mobility patterns and pre-allocate resources, ensuring smoother transitions during handovers. Multiaccess handover strategies are also being implemented to maintain session continuity and reduce latency during mobility events.

Security vulnerabilities further complicate the dependability of MEC and IoT systems [2]. Edge devices are often deployed in physically insecure environments, making them susceptible to tampering, theft, and physical damage. Cyberattacks, including Distributed Denial-of-Service (DDoS) and man-in-the-middle (MITM) attacks, pose significant threats to data integrity, availability, and confidentiality. Additionally, the decentral- ized nature of edge computing increases the attack surface, leaving multiple entry points for attackers. Privacy concerns are also prevalent, as sensitive data processed at the edge may be intercepted during transmission. To strengthen security and dependability, advanced encryption methods, secure boot mechanisms, and anomaly detection systems are being inte- grated into MEC architectures. Blockchain technology is also gaining attention as a means of ensuring data integrity and building trust between interconnected devices [2]. Addressing dependability challenges in MEC and IoT systems requires a multi-faceted approach that combines resource optimization, fault-tolerant mechanisms, interoperability frameworks, mobil- ity management techniques, and robust security measures. As these technologies continue to evolve, innovative solutions will be crucial in enhancing the reliability and performance of edge computing systems.

B. Dependability of Hardware Performance Counters in Embedded Systems

Hardware Performance Counters (HPCs) are vital compo- nents in embedded systems, enabling performance monitoring, debugging, and optimization by tracking metrics such as instruction counts, cache misses, and pipeline stalls. Despite their importance, ensuring the dependability [5] of HPCs in embedded systems poses several challenges due to resource limitations, susceptibility to faults, and security vulnerabili- ties. Embedded systems often operate under strict resource constraints, including limited processing power, memory, and energy availability, which makes implementing complex de- pendability mechanisms difficult. These limitations result in a trade-off between performance and reliability, especially in applications requiring real-time responses, such as automotive systems, industrial automation, and medical devices.

A significant challenge in ensuring dependability is counter overflow and data corruption [5]. HPCs typically have fixed- width registers, which can overflow during intensive opera- tions, leading to inaccurate data collection and compromised monitoring. Such inaccuracies can impact performance evalua- tions and fault diagnosis, potentially leading to system failures in safety-critical applications. Additionally, embedded systems often face environmental stresses, such as temperature varia- tions, vibrations, and electromagnetic interference [5], further affecting the reliability of HPCs. To address these issues, fault-tolerant mechanisms like error detection codes, parity checks, and checkpointing techniques are employed to monitor and recover from errors effectively. These approaches enhance system resilience without imposing significant computational overhead.

Another critical concern is real-time constraints in embedded environments. Embedded systems are often deployed in sce- narios where timing precision is crucial, such as autonomous vehicles and avionics systems. Delays caused by faulty coun- ters or performance monitoring overhead can violate timing re- quirements, leading to system instability. Dependable solutions must therefore ensure low-latency monitoring while maintain- ing high accuracy and consistency in performance metrics. Approaches such as hardware-assisted monitoring frameworks and lightweight profiling techniques [5]can provide real-time performance insights without sacrificing responsiveness.

Security vulnerabilities also pose a significant threat to HPC dependability. HPCs are susceptible to side-channel attacks, where attackers exploit counter data patterns to extract sensi- tive information, such as encryption keys. Embedded systems deployed in critical infrastructure or IoT networks are espe- cially vulnerable to such attacks due to their distributed and often insecure nature. Moreover, unauthorized access to HPCs can lead to tampering with performance data, affecting the in- tegrity of monitoring and analysis processes. To address these issues, secure counter designs with access control policies, encryption mechanisms, and isolation techniques are essential for protecting sensitive data and ensuring the reliability ofHPCs. The increasing adoption of virtualization in embedded systems introduces additional dependability challenges. Virtualization improves resource utilization and scalability but also requires effective isolation mechanisms [5] to prevent interference between virtual machines (VMs). Shared HPCs across VMs may lead to performance measurement inaccuracies and data leaks, compromising dependability. To counter these chal- lenges, hypervisor-level monitoring frameworks are being de- veloped to ensure accurate and isolated performance tracking. Additionally, redundant designs and replicated counters are employed to mitigate failures caused by virtualization overhead or resource contention.

Achieving high dependability in hardware performance coun- ters for embedded systems requires a multi-faceted approach. This includes error detection and correction techniques, redun- dancy mechanisms, and lightweight fault-tolerant frameworks to handle failures efficiently. Furthermore, real-time profiling tools, secure architectures, and virtualization-aware monitoring are crucial to protect performance counters from security threats and ensure consistent, accurate performance measure- ments [5]. These advancements are critical to supporting the growing demand for reliability and resilience in modern embedded systems, particularly in safety-critical and mission- critical applications.

IV. PERFORMANCE TRADE-OFFS

A. Performance metrics and trade-offs in 5G MEC and IoT systems.

Performance in 5G Multi-Access Edge Computing (MEC) and IoT systems is influenced by several trade-offs involving latency, energy efficiency, scalability, and resource utilization. One of the key performance metrics is latency, which must be minimized to support time-sensitive applications such as autonomous vehicles, augmented reality, and industrial automation. MEC reduces latency by processing data closer to the end user, but this comes at the cost of resource constraints due to limited computational and storage capabilities at edge nodes. Ensuring low latency often requires distributed processing and load balancing mechanisms, which may introduce additional overhead and complexity. Another crucial metric is energy efficiency, particularly in IoT systems where devices often operate on battery power. While offloading computations to MEC nodes can reduce energy consumption on IoT devices, it introduces network transmission costs that may offset energy savings. Optimizing this trade-off involves designing adaptive offloading algorithms and lightweight communication protocols to minimize energy consumption without compromising performance. Scalability is also a concern, as MEC and IoT systems need to handle a growing number of connected devices and dynamic workloads. Achieving scalability requires dynamic resource allocation and container-based virtualization, [5] but these techniques can increase latency and reduce reliability under high loads. Security and privacy requirements further impact performance. Implementing encryption and anomaly detection mechanisms in MEC environments ensures data integrity and confidentiality but increases computational overhead and latency. Similarly, deploying machine learning algorithms for real-time threat detection in IoT systems enhances security but may strain resource-constrained devices. Addressing these trade-offs involves integrating hardware accelerators, such as GPUs and FPGAs, to improve processing power and reduce overhead without compromising security. [5]

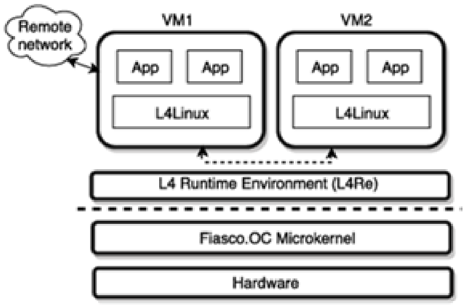

Fig. 3. Usecase of multipe virtual machines [5]

1) **Hardware:** This is the physical foundation of the system, such as the CPU, memory, and other hardware components.
2) **Fiasco.OC Microkernel:** This is a small, efficient microkernel that provides basic services like memory management, process scheduling, and inter-process communication (IPC). It sits directly on top of the hardware.
3) **L4 Runtime Environment (L4Re):** This layer provides a higher-level abstraction on top of the Fiasco.OC microkernel. It offers services like virtual memory management, device drivers, and network support.
4) **VM1 and VM2:** These represent virtual machines (VMs) running on the L4Re. Each VM has its own isolated environment with its own operating system (in this case, L4Linux).
5) **Applications (App):** These are the end-user applications running within each VM.
6) **Remote Network:** This represents a network connection that allows VMs to communicate with other systems or the outside world.

Key Concepts:

- **Microkernel Architecture:**The L4 microkernel is a minimalistic kernel that provides only essential services. This design approach is known as microkernel architecture, which aims to improve system security, flexibility, and modularity.
- **Virtualization:** The L4Re allows for the creation of multiple isolated VMs on a single hardware platform, enabling efficient resource utilization and improved security.
- **L4Linux:** This is a Linux-based operating system specifically designed to run on the L4 microkernel. It provides a familiar Linux environment for applications while leveraging the benefits of the L4 microkernel architecture. In essence, the image depicts a system where multiple VMs, each running its own operating system and applications, are hosted on a single hardware platform through the L4 microkernel and its runtime environment. This architecture offers advantages like improved security, flexibility, and resource management compared to traditional monolithic operating systems.

B. Hardware and Virtualization Performance Issues in Embedded Systems and SDN

In embedded systems, performance trade-offs are shaped by hardware limitations, real-time constraints, and virtualiza- tion overheads. Hardware performance counters (HPCs) are commonly used to monitor system performance, but their effectiveness is limited by counter overflow, data corruption, and resource contention. Embedded systems often prioritize low power consumption and compact designs, which restrict the use of high-performance [5] processors and extensive memory resources. As a result, achieving high performance often comes at the expense of reliability, as error detection and fault-tolerant mechanisms introduce additional computational overhead.

Virtualization further complicates performance in embedded systems. While virtualization enhances flexibility and resource sharing, it introduces latency and isolation issues, especially when multiple virtual machines (VMs) compete for shared resources. Performance degradation occurs due to context switching and I/O bottlenecks, which impact real-time op- erations. To mitigate these effects, lightweight virtualization techniques [7], such as microkernels and hypervisors, are employed. These approaches improve isolation and reduce overhead but may still struggle to meet the strict timing constraints required by embedded applications.

In Software-Defined Networks (SDN), virtualization enables flexible network management and scalability but also in- troduces latency and performance bottlenecks. Virtualized network functions (VNFs) [5] and software-based switching mechanisms are slower than traditional hardware switches, affecting throughput and reliability. Security overhead in SDN environments, such as encryption and traffic monitoring, further impacts performance. The trade-off between performance and security is particularly evident in multi-tenant SDN architectures, where ensuring data isolation and integrity often reduces network efficiency.

To address these issues, hardware accelerators, such as Field- Programmable Gate Arrays (FPGAs) and Graphics Processing Units (GPUs) [7], are increasingly being used to enhance performance in embedded and SDN environments. These ac- celerators improve processing speed and reduce virtualization overhead but require specialized programming models and in-crease deployment costs. Additionally, container-based virtual- ization is emerging as an alternative to traditional hypervisors, offering faster startup times and lower resource consumption. However, containers may provide weaker isolation, making them more vulnerable to security threats.

V. ROLE OF MACHINE LEARNING IN ENHANCING SECURITY

A. Applications and implications of machine learning for IoT and smart grid security

Machine learning (ML) has emerged as a transformative technology for improving security in IoT systems and smart grids by enabling advanced threat detection, anomaly identi- fication, and predictive analytics. Given the complex and dy- namic nature of these systems, traditional security approaches often struggle to address evolving threats. ML techniques offer adaptive and scalable solutions by learning patterns, detecting deviations, and responding to security breaches in real-time. In IoT systems, ML algorithms are widely used for intrusion detection and malware classification [3]. IoT devices often operate in distributed and resource-constrained environments, making them susceptible to cyberattacks, such as Distributed Denial-of-Service (DDoS) attacks, spoofing, and data in- jection. ML models, including supervised learning methods like Support Vector Machines (SVMs) and Decision Trees, as well as unsupervised approaches such as clustering and anomaly detection, can analyze network traffic patterns to detect abnormal behavior. These models continuously learn from incoming data, allowing them to adapt to new attack strategies [6]. For example, deep learning algorithms, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have demonstrated high accuracy in identifying malware signatures and detecting zero-day attacks without prior knowledge.

In the context of smart grids, ML enhances both security and privacy by monitoring power consumption data and predicting unusual patterns indicative of energy theft, cyberattacks, or equipment failures. Smart grids rely heavily on communica- tion networks and distributed sensors, which expose them to vulnerabilities such as data manipulation and remote control attacks. ML-based anomaly detection systems analyze histori- cal and real-time data to identify irregularities, enabling early detection and prevention of malicious activities. Techniques such as Random Forests and Neural Networks are employed to classify threats, while reinforcement learning is used to optimize the response strategies for mitigating attacks [3].

One significant implication of applying ML in security systems is its ability to handle big data generated by IoT devices and smart grids. ML algorithms are capable of processing vast amounts of data in real-time, identifying correlations, and recognizing complex patterns that may be missed by traditional methods. For example, ensemble learning tech- niques combine multiple models to improve accuracy and reduce false positives, ensuring more reliable threat detection. Additionally, federated learning [6] has gained attention as a privacy-preserving ML approach, where models are trained locally on IoT devices without transferring sensitive data to centralized servers, thus enhancing data privacy.

However, integrating ML into IoT and smart grid security also introduces challenges and trade-offs. ML models require extensive training datasets, and their performance heavily depends on data quality and diversity. Adversarial attacks, where attackers manipulate input data to mislead ML models, pose a significant risk to security systems. For instance, an attacker could alter sensor readings in a smart grid to by- pass anomaly detection systems. Addressing these challenges requires the development of robust ML algorithms [3] that can resist adversarial manipulations and self-adapt to evolving threats. Moreover, the resource limitations of IoT devices often constrain the deployment of complex ML models. To overcome this, lightweight ML techniques, such as compressed neural networks and edge AI, are being employed to perform computations directly on edge devices, reducing dependency on centralized servers and enhancing scalability. For smart grids, integrating ML with blockchain technology further improves security by ensuring tamper-proof data storage and secure transactions.

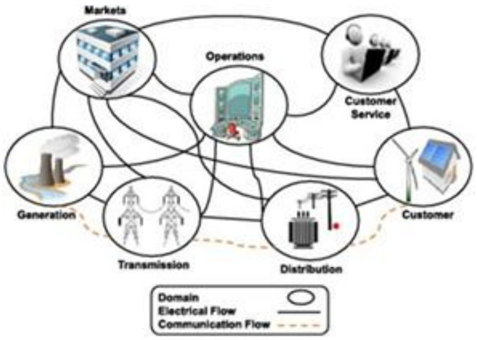

Fig. 4. Modified NIST smart grid architecture [5]

1. **Markets:**This domain encompasses the various markets and mechanisms for buying, selling, and trading electricity. It includes entities like energy retailers, wholesale markets, and demand response programs.
2. **Operations:** This domain focuses on the operational aspects of the grid, such as monitoring, control, and management of the electrical system. It includes entities like system operators, grid controllers, and SCADA systems.
3. **Customer:** This domain represents the end-users of electricity, including residential, commercial, and industrial customers. It also includes entities like customer service centers and customer information systems.
4. **Generation:** This domain encompasses all sources of electricity generation, including traditional sources like coal and natural gas power plants, as well as renewable sources like wind and solar power.
5. **Transmission:**This domain refers to the high-voltage power lines that transmit electricity over long distances from generation sources to distribution centers.
6. **Distribution:** This domain refers to the lower-voltage power lines that distribute electricity from distribution centers to individual customers.

B. Significance of the NIST Smart Grid Framework:

The NIST Smart Grid Framework provides a valuable tool for understanding the complexity of smart grid systems. It helps stakeholders, including utilities, regulators, and policymakers, to visualize the different components and interactions within the grid. This framework can be used to:

- **Identify interdependencies:**Understand how different components of the grid are interconnected and how changes in one domain can impact others.
- **Develop standards and guidelines:**Establish common standards and guidelines for the development and deployment of smart grid technologies.
- **Assess risks and vulnerabilities:** Identify potential risks and vulnerabilities within the grid and develop strategies to mitigate them.
- **Promote innovation:** Encourage innovation in smart grid technologies by providing a clear understanding of the system architecture and requirements.

VI. COUNTERMEASURES AND PROPOSED SOLUTIONS

Ensuring security and dependability in 5G MEC, IoT sys- tems, smart grids, embedded systems, and Software-Defined Networks (SDN) requires robust frameworks and innovative solutions. The papers reviewed propose various approaches, addressing challenges such as latency, resource constraints, fault tolerance, and cybersecurity vulnerabilities. These solu- tions emphasize improving resilience, scalability, and perfor- mance across different paradigms.

A. Security Frameworks and Dependability Enhancements

1) **5G MEC and IoT Systems Security Frameworks:** In MEC and IoT environments, the focus is on creating distributed security frameworks that address data privacy and communication vulnerabilities [2]. Machine Learning (ML)-based intrusion detection systems (IDS) are widely proposed to detect malicious activities in real-time. These frameworks use anomaly detection algorithms and behavioral analytics to identify attacks, such as Distributed Denial-of-Service (DDoS) and spoofing [3].
2) **Dependability Enhancements:** MEC architectures are enhanced through fault-tolerant designs, including redundancy mechanisms and self-healing networks that recover from failures automatically. Additionally, resource allocation algorithms dynamically balance workloads, improving latency and throughput without compromising reliability [3]
3) **Embedded Systems:** Security Frameworks:For embedded systems, the proposed solutions involve lightweight cryptography and hardware-based isolation techniques to secure performance counters and prevent side-channel attacks. Virtualization frameworks, such as microkernels, improve security by ensuring process isolation and protecting against data leakage between virtual machines [5].

4) **Smart Grids:** Security Frameworks: Smart grid sys- tems benefit from ML-based threat detection systems, which monitor power flow patterns and predict cyber- attacks, such as energy theft and data manipulation [6]. Blockchain integration is proposed as an additional security layer, enabling tamper-proof data storage and secure transactions between devices.
5) **Dependability Enhancements:** To address dependability, distributed energy management systems are designed, incorporating load balancing algorithms [6] and resilient communication protocols that maintain stability even during attacks or network disruptions.
6) **SDN:** Security Frameworks: In SDN environments, proposed frameworks emphasize security policies enforced through centralized controllers. These frameworks monitor traffic flows and dynamically adapt rules to mitigate threats [7]. Additionally, virtualized network functions (VNFs) use encrypted communications and access control mechanisms to enhance security.
7) **Dependability Enhancements:** SDN architectures adopt redundancy and failover mechanisms to ensure high availability during network failures [7]. Load bal- ancing algorithms and traffic engineering techniques further optimize performance while maintaining depend- ability.

B. Research Gaps and Future Directions

Despite significant advancements in security, dependability, and performance optimization across paradigms like 5G MEC, IoT systems, smart grids, embedded systems, and SDN, several unresolved challenges remain. Addressing these gaps is crucial for enabling scalable, adaptive, and resilient systems. This section highlights the key research gaps identified in the reviewed papers and proposes future research directions for integrating security, dependability, and performance in these paradigms.

VII. UNRESOLVED CHALLENGES

1) Security Challenges: Adversarial Attacks on Machine Learning Models: While ML-based frameworks improve threat detection, they remain vulnerable to adversarial attacks where malicious inputs deceive the model. De- veloping robust ML algorithms resistant to such attacks is still an open area of research.

1. **Privacy-Preserving Techniques for IoT and Smart Grids:** Ensuring data privacy in resource-constrained devices remains challenging, espe- cially during data transmission and processing. Existing frameworks often lack scalable solutions to support end-to-end encryption and federated learning without introducing excessive overhead.
2. **Scalable Blockchain Solutions:** Blockchain enhances security and integrity in IoT networks and smart grids, but its latency and scalabil- ity issues hinder deployment in large-scale envi- ronments. Lightweight, energy-efficient consensus mechanisms need further exploration.
3. **Dynamic Security Management in SDN:** SDN faces challenges related to centralized vulnerabilities and real-time attack mitigation. The dependency on central controllers poses risks if they are compromised, requiring distributed security frameworks for SDN environments.

2) Dependability Challenges:

1. **Fault Tolerance in MEC and IoT:** Ensuring fault recovery in highly dynamic and distributed MEC environments is complex, particularly with heterogeneous hardware and frequent node mobility. Existing fault-tolerant mechanisms need improvements to handle multi-failure scenarios effectively.
2. **Resource Optimization for Embedded Systems:** Embedded systems face trade-offs between dependability and performance due to hardware constraints. Research is needed to develop adaptive resource allocation techniques that ensure consistent performance while minimizing energy consumption.
3. **Reliability of Virtualization Frameworks:** Virtualization in embedded systems and SDN improves flexibility but introduces performance degradation due to context switching and I/O overheads. Enhancing real-time performance and resource isolation requires further investigation.

3) Performance Challenges:

1. **Latency-Performance Trade-offs in MEC and IoT:** Achieving low-latency processing without sacrificing performance under scalable workloads remains a bottleneck. Existing solutions struggle with balancing real-time requirements and com- putational complexity in resource-limited environ- ments.
2. **Hardware-Accelerated Security Mechanisms:** While hardware performance counters (HPCs) improve monitoring and security in embedded systems, their integration with ML models and virtualization frameworks introduces performance overheads. Further research is required lightweight accelerators and energy-efficient processors.
3. **SDN Performance Optimization:** SDN relies on virtualized network functions (VNFs), which often

face throughput limitations. Techniques to optimize packet processing and load balancing need further exploration to meet the demands of modern applications.

VIII. FUTURE RESEARCH DIRECTIONS

1. **Integrating Security, Dependability, and Performance:**
   a) **AI-Driven Security Architectures:** Future re- search should focus on designing AI-based adap- tive security frameworks that dynamically detect and mitigate threats while maintaining system per- formance. These frameworks can leverage reinforcement learning to optimize resource allocation and improve resilience.
   b) **Federated and Transfer Learning Models:** Ex- panding the use of federated learning and transfer learning can improve data privacy and scalability in IoT and smart grids without requiring centralized data storage. Future studies should evaluate these approaches under heterogeneous environments.
2. **Lightweight Cryptography and Blockchain Innovations:** Developing lightweight encryption techniques and blockchain-based authentication protocols can pro- vide scalable and tamper-proof security for resource- constrained devices. Research should focus on hybrid blockchain frameworks that balance decentralization and efficiency for IoT ecosystems.
3. **Fault-Tolerant Resource Management:**Introducing AI-enhanced fault detection and self-healing systems can improve reliability in MEC and SDN infrastructures. Research can explore distributed decision-making algo- rithms that reduce downtime and enable predictive fault management.
4. **Hardware-Software Co-Design:**Future work should emphasize hardware-software integration for optimizing virtualization performance in embedded systems. De- signing custom hardware accelerators to support ML inference and real-time monitoring can help address latency and energy constraints.
5. **Scalable and Dynamic SDN Frameworks:** SDN architectures need improvements in dynamic security policies and distributed control mechanisms to mitigate centralized vulnerabilities. Research should explore blockchain-based SDN controllers and programmable data planes to enhance scalability and fault tolerance.
6. **Cross-Domain Integration and Interoperability:** As systems become increasingly interconnected, future re- search must address interoperability challenges across paradigms. Designing unified frameworks that integrate MEC, IoT, smart grids, and SDN with common security protocols and data exchange standards will be critical.

IX. CONCLUSION

The integration of security, dependability, and performance optimization across paradigms such as 5G MEC, IoT systems, smart grids, embedded systems, and SDN is essential for ensuring robust, scalable, and resilient infrastructures. While significant advancements have been made in each of these ar- eas, several challenges remain unresolved, particularly in terms of adapting to new attack vectors, maintaining performance under resource constraints, and achieving fault tolerance in dynamic environments.

Security frameworks are increasingly relying on machine learning and blockchain technologies to enhance intrusion de- tection, data privacy, and tamper-proof data storage. However, these approaches still face challenges related to scalability, latency, and the vulnerability of AI models to adversarial attacks. Similarly, ensuring dependability remains a significant challenge, particularly with heterogeneous infrastructures and the need for fault recovery mechanisms in real-time systems like IoT and smart grids.

Performance trade-offs must be carefully managed, partic- ularly in resource-constrained environments like embedded systems, where maintaining low-latency processing without sacrificing reliability or security is a constant balancing act. Virtualization techniques and hardware performance counters offer potential solutions but often introduce performance over- heads. Looking ahead, future research directions should focus on integrating AI-driven security frameworks that can adapt dynamically to emerging threats while maintaining system per- formance. Further exploration into lightweight cryptography, federated learning, and blockchain-based solutions will also be crucial in addressing privacy and scalability challenges. Addi- tionally, enhancing hardware-software integration for real-time fault detection and performance optimization will be essential for ensuring that these systems can meet the demands of modern applications.

In conclusion, while the advancements in security, depend- ability, and performance optimization across these paradigms have laid a strong foundation, ongoing research is needed to overcome existing challenges and design more efficient, adaptive, and resilient systems for the future. Addressing these research gaps will be key to enabling the next generation of 5G, IoT, smart grids, and SDN infrastructures.

REFERENCES

[1] T. Alharbi and M. Portmann, “The (in)security of virtualization in software defined networks,” *IEEE Access*, vol. 7, pp. 66584–66594, 2019.

[2] G. Nencioni, R. G. Garroppo, and R. F. Olimid, “5g multi-access edge computing: A survey on security, dependability, and performance,” *IEEE Access*, vol. 11, pp. 63496–63533, 2023.

[3] M. Bagaa, T. Taleb, J. B. Bernabe, and A. Skarmeta, “A machine learning security framework for iot systems,” *IEEE Access*, vol. 8, pp. 114066– 114077, 2020.

[4] I. Ahmad, J. Pinola, I. Harjula, J. Suomalainen, E. Harjula, J. Huusko, and T. Kumar, “An overview of the security landscape of virtual mobile networks,” *IEEE Access*, vol. 9, pp. 169014–169030, 2021.

[5] D. Mathew, B. A. Jose, J. Mathew, and P. Patra, “Enabling hardware performance counters for microkernel-based virtualization on embedded systems,” *IEEE Access*, vol. 8, pp. 110550–110564, 2020.

[6] P. Haji Mirzaee, M. Shojafar, H. Cruickshank, and R. Tafazolli, “Smart grid security and privacy: From conventional to machine learning issues (threats and countermeasures),” *IEEE Access*, vol. 10, pp. 52922–52954, 2022.

[7] P. Haji Mirzaee, M. Shojafar, H. Cruickshank, and R. Tafazolli, “Smart grid security and privacy: From conventional to machine learning issues (threats and countermeasures),” *IEEE Access*, vol. 10, pp. 52922–52954, 2022.

[8] N. Borgioli, M. Zini, D. Casini, G. Cicero, A. Biondi, and G. Buttazzo, “An i/o virtualization framework with i/o-related memory contention control for real-time systems,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 11, pp. 4469– 4480, 2022.

[9] Y. Liu, Y. Chen, Y. Jiao, H. Ma, and T. Wu, “A shared satellite ground station using user-oriented virtualization technology,” *IEEE Access*, vol. 8, pp. 63923–63934, 2020.

[10] Z. Chen, W. Dong, H. Li, P. Zhang, X. Chen, and J. Cao, “Collaborative network security in multi-tenant data center for cloud computing,” *Tsinghua Science and Technology*, vol. 19, no. 1, pp. 82–94, 2014.

[11] J. Wang, S. Hao, H. Hu, B. Zhao, H. Li, W. Zhang, J. Xu, P. Liu, and J. Ma, “S-blocks: Lightweight and trusted virtual security function with sgx,” *IEEE Transactions on Cloud Computing*, vol. 10, no. 2, pp. 1082– 1099, 2022.