Article

Towards Double Defense Network Security Based on Multi-Identifier Network Architecture

Yunmin Wang [1], Abla Smahi [1], Huayu Zhang [2] and Hui Li [1,3,*]

[1] School of Electronic and Computer Engineering, Peking University, Shenzhen 518055, China; wangyunmin@pku.edu.cn (Y.W.); smahi_abla@pku.edu.cn (A.S.)
[2] Purple Mountain Laboratories, Nanjing 211111, China; 1301111205@sz.pku.edu.cn
[3] Peking University Lab of China Environment for Network Innovation, Peking University, Shenzhen 518055, China
* Correspondence: lih64@pkusz.edu.cn

Abstract: Recently, more and more mobile devices have been connected to the Internet. The Internet environment is complicated, and network security incidents emerge endlessly. Traditional *blocking and killing* passive defense measures cannot fundamentally meet the network security requirements. Inspired by the heuristic establishment of multiple lines of defense in immunology, we designed and prototyped a Double Defense strategy with Endogenous Safety and Security (DDESS) based on multi-identifier network (MIN) architecture. DDESS adopts the idea of a zero-trust network, with identity authentication as the core for access control, which solves security problems of traditional IP networks. In addition, DDESS achieves individual static security defense through encryption and decryption, consortium blockchain, trusted computing whitelist, and remote attestation strategies. At the same time, with the dynamic collection of data traffic and access logs, as well as the understanding and prediction of the situation, DDESS can realize the situation awareness of network security and the cultivation of immune vaccines against unknown network attacks, thus achieving the active herd defense of network security.

Keywords: network security; double defense; zero trust; situation awareness; immunology

check for updates

Citation: Wang, Y.; Smahi, A.; Zhang, H.; Li, H. Towards Double Defense Network Security Based on Multi-Identifier Network Architecture. *Sensors* **2022**, *22*, 747. https://doi.org/10.3390/s22030747

Academic Editors: Peter Han Joo Chong and Omprakash Kaiwartya

Received: 15 December 2021
Accepted: 17 January 2022
Published: 19 January 2022

1. Introduction

With the development of the Internet and its deep integration with human social life, more and more mobile devices are connected. They are heterogeneous and ubiquitous, and put forward higher requirements for network security. The predominant *best-effort* design of TCP/IP network architecture focuses on the end-to-end communication between non-commercially and mutually trusted users. The network data transmission is entirely transparent, and the service and the bearer are separated. However, it instead hands over the security control to the users. This helps to improve the network availability and flexibility but at the cost of network security [1]. The Internet appears to be incapable of responding to users' demands for security in obtaining massive content. In addition, the current internet IP addresses have positioning, identity, and forwarding functions that pose many challenges for supporting quality of service (QoS), especially for the Internet of things (IoT) [2], Internet of Vehicles (IoV) [3], and the authenticity and credibility of the fusion of the virtual and real world in the metaverse [4].

Currently, network security protection is mainly based on IP networks' characteristics and adopts passive defense measures with *blocking and killing* methods, such as firewalls, authentication technology, access control, vulnerability scanning, disaster recovery, and honeypot technology. The defense capabilities of these measures can be passive or static, depending on predetermined settings before accessing the system and updating the preset defense library during use. Therefore, they can only detect and defend against a number of predefined network security attacks. Moreover, although these methods require higher user permissions and privileges, they can be easily controlled and exploited by attackers [5]. It is worth mentioning that the network security vulnerabilities constantly emerge while the attack methods are persistently refreshed. As demonstrated in Figure 1, attackers usually scan the weakest link in security defense to launch network attacks. Traditional defense measures focus on improving the protection capabilities against attacks rather than identifying, tracking, and investigating the responsibility of the attackers. They passively receive every intrusion attack, which is difficult to detect, identify, and respond to emerging attack methods, and it is challenging to solve network security problems fundamentally.

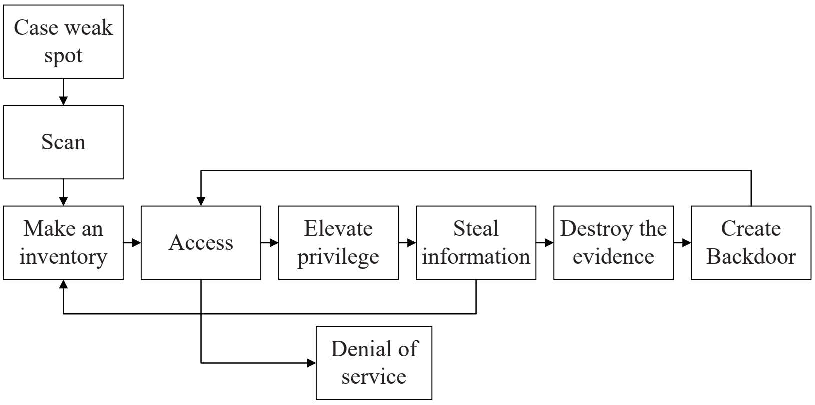


Figure 1. Network attack procedures.

While paying attention to network security, we found much inspiration from the biological immune system. The biological immune system has the advantages of feature extraction, distributed detection, self-tolerance, self-adaptation, robustness, and the capabilities of pattern recognition, learning, and memory [6]. It forms individual immunity to viral infections through its physical barrier, innate immune system, and adaptive immune system, and achieves the public herd immunity through collaboration between heterogeneous nodes and the cultivation and injection of vaccines [7]. Inspired by the multiple defense lines in immunology [8], this paper proposes a double defense strategy with endogenous safety and security (DDESS) as shown in Figure 2 based on the multi-identifier network (MIN) architecture [9]. We adopt identity authentication as the core access control method to solve traditional IP network security problems, and implement static network security defense through key encryption technology, blockchain technology, and trusted computing whitelist strategy. At the same time, through the dynamic collection of data traffic and access logs, DDESS can find the network vulnerabilities in the existing system, understand and predict the situation, realize the situation awareness of network attacks, and cultivate the immune vaccine of unknown network attacks, to complete the active dynamic defense of network security and the balance of network availability and security.

In summary, this paper makes the following contributions:

- This paper discusses the problem of the network's bottleneck resulting from the traditional IP-network carrying capacity and the poor network security. In this regard, we inspire our proposed system from the idea of zero-trust network [10]. In doing so, based on MIN architecture [9,11], we built a security strategy with an identity identifier as the core of network transmission. This strategy is built based on identity identifiers rather than network locations. Only after user authentication, authorization, and account verification can the user and the application communicate.
- Signature of network transmission packets protects data from theft. A variety of the data related to the identifiers is stored in the consortium blockchain of the MIN to ensure the identifier data's non-repudiation, tampering-resistance, and traceability. The independently developed voting consensus algorithm Proof of Vote (PoV) [12] improves the throughput of the system. At the same time, the encryption key and data

are stored in the trusted computing modules (TCM) to ensure that they are not read or disclosed [13], and protected by remote authentication [14]. Real-time monitoring of the trusted whitelist of accessing applications can promptly detect and respond to attacks.

- The network traffic and application access logs are collected and analyzed for further situation awareness. DDESS classifies and extracts appropriate attack information, including attack behavior and mode, and makes use of the convolutional neural networks (CNN) to evaluate the network security attack index of attack information. In addition, DDESS predicts the network security of the existing network system and provides the corresponding solutions. DDESS simulates network behaviors in the sandbox. It then analyzes these simulation results, cultivates a security immune vaccine, enriches the network attack behavior database, and prevents unknown network attack behaviors.

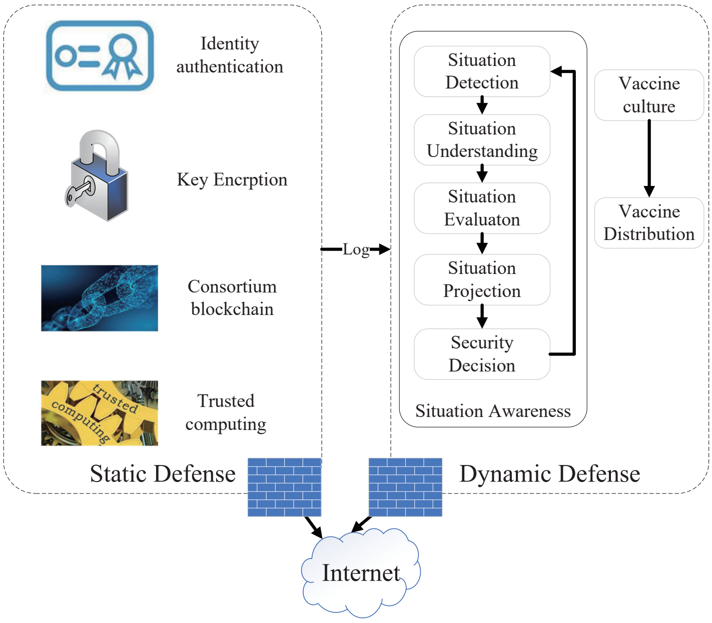


Figure 2. Double defense strategy with endogenous safety and security (DDESS).

The remainder of this paper is organized as follows: Section 2 reviews the related work of image anomaly detection. A detailed description of our proposed method is given in Section 3. In Section 4, we present experimental setups and comparisons. Section 5 carries out the corresponding experiments and gives out the simulations results. Section 6 concludes the paper and points out some of our future research.

2. Related Work

The traditional static defense of network security is divided into three types: reinforcement protection of the system, intrusion detection, and network deception [15].

Firewall (including packet filtering, proxy type, state inspection, in-depth inspection, web application) [16], encryption and decryption, data authentication [17], and access control [18] focus on protecting information and enhancing the security of the network system itself. They play a protective role in ensuring the normal access channels of the network system, authenticating legitimate user identities and rights management, and the security of confidential data and information.

Intrusion detection [19], vulnerability detection [20], traffic analysis [21], log auditing [22], and other attack methods for known characteristic information make use of methods such as characteristic scanning, pattern matching, and comprehensive data analysis to conduct dynamic monitoring, linkage alarms, and emergency response to prevent or eliminate attack threats.

Honeypot technology [23] deploys some hosts and network services as decoys to induce attackers to carry out attacks, thereby capturing and analyzing the attack behaviors, understanding the tools and methods used by the supplier, and speculating the intention and motivation of the attack, thereby enhancing its security protection capabilities [24]. However, it is difficult to deploy a honeypot environment that is not readily perceivable by intruders [25].

The above static defense methods can better defend against attacks with known characteristics and fixed patterns. However, it cannot defend against attacks based on unknown vulnerabilities backdoors, complicated and changeable multimode joint attacks, and attacks from within the network. With the continuous improvement of the automation and intelligence of countermeasures, the construction and strengthening of the network security defense system alone can no longer meet the actual needs of network security defense. Dynamic defense technology has gradually attracted widespread attention and is considered a revolutionary technology that changes the asymmetry of network security, such as Moving Target Defense (MTD) [26], Cyberspace Mimic Defense (CMD) [27], and Cyber Deception (CD) [28].

Let us dive into a specific scenario: vulnerabilities have constantly been discovered in the Internet of Vehicles (IoV), and the security problems of Vehicle to Everything (V2X) interactive communication are gradually emerging. The Internet of Vehicles requires real-time and timely resolution of network security issues. Therefore, many machine learning-based solutions in V2X scenarios have been proposed to provide dynamic defenses and address these problems. The surveillance of physical layer security (PLS) was explored in the field of connected vehicles [29]. A delimitated anti jammer scheme based on machine learning [30] secures the network and alleviates the traffic congestion simultaneously; in the meantime, it can reduce the computing delay.

Network dynamic defense is an innovative network defense technology system gradually developed to deal with the increasingly severe cyberspace security situation, which makes it possible to break the long-standing impregnable asymmetry and will balance the difficulty of network attack and defense in the future.

In this paper, we present a network defense strategy that integrates static and dynamic defenses. It adopts the ideas of the zero-trust network, and employs identity authentication, blockchain technology, and trusted computing technology, with situation awareness and dynamic immune functions.

3. Static Defense

We adopt the idea of a zero-trust network and build a multi-identifier network system with identity as the core, supplemented by data signature, blockchain technology, and trusted computing technology to realize the static defense of network security to improve the autoimmunity of network individuals.

3.1. Multi-Identifier Network System with Identity as the Core

The network identifier is the data carried in the network packet for addressing and forwarding by the intermediate router. In the traditional IP network, the target IP address of a network packet is its network identifier, forming a thin waist hourglass structure with IP as the core in the network layer, which identifies two network nodes for end-to-end communication. This network architecture is no longer suitable for the network communication requirements of the existing network to obtain content and services. At the same time, since the original design purpose of the IP network is to communicate between non-commercial users who trust each other, the network transmission is entirely transparent to the intermediate router, which makes data theft and monitoring easy. The IP network lacks top-level design and usually adopts a passive patching method for network security, making it increasingly challenging to implement [31].

The multi-identifier network (MIN) system [11] takes identity as the core, and supports the coexistence of multiple network identifiers such as identity, content [32], service, space,

space, and location information, and IP, to solve the depletion of IP addresses and the security problems existing in IP networks, as shown in Figure 3. The multi-identifier network is divided into a management plane (multi-identifier system, MIS) and data plane (multi-identifier routers, MIR), which supports simultaneous transmission of multiple network identifiers in the network, as depicted in Figure 4. MIS supports the management and resolution of network identifiers by multiple parties equally, and is responsible for the affairs related to identifiers combined with offline. Its main functions include identifier registration, identifier query, identifier generation, and identifier query. The multi-identifier router (MIR) is mainly responsible for the addressing, forwarding, and mutual translation functions of multiple identifier network packets. It supports the push transmission mode of identity and IP identifiers and the pull transmission mode of content identifier and service identifier.

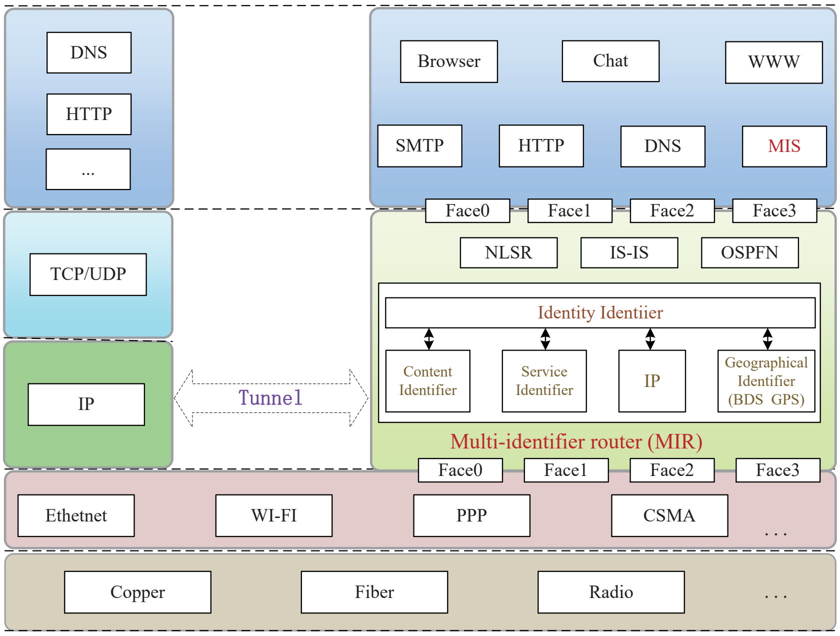


Figure 3. Multi-identifier network architecture.

Identity is not limited to users but also includes the unique identification of network communication physical entities (from now on referred to as users) such as devices, interfaces, applications, and business systems in the digital network world. It is the equivalent of physical network entities in the digital network world. The identity of a physical entity is unique in the digital network world, which is different from a username. A physical entity can have different usernames in different systems.

The identity identifier is the core network identifier in the MIN, and all multi-identifier routers need to support routing and forwarding of the identity identifier. Other types of network identifiers will be associated with a particular identity identifier. In an unsupported network domain, it can go back to the identity identifier for data forwarding. The hash value of the real identity information of the network communication entity (such as ID number, fingerprint, face, voiceprint, iris, and other biometric information, password) is included in the signature field in the data packet. It carries the sender's accurate identity signature information and time stamps, realizes dynamic access control based on identity, establishes a unified digital identity identifier and life cycle management for users and other physical entities participating in network communications, and saves them in the consortium blockchain of the MIS system.

After the communication entity is registered in the MIS, a unique identity is formed, corresponding permissions are assigned according to its role identification, and fine-grained permission management is carried out by monitoring its access behavior to realize the management and control of dynamic permissions. The principle of the least authority is implemented for communication entities, and their authority is allocated reasonably to ensure that each communication entity (including applications, services, users) can only access the required information or resources. When it is necessary to access sensitive resources, the identity recognition module will call other real identity information saved in the MIS for secondary verification, such as SMS verification code, dynamic password, face verification, to form access control and dynamic threat identification, privilege confirmation, alarm, and blocking at the regional boundary. The information transferred by users in the system will be recorded. The data package is highly bound with the user identity information to realize the traceability of data and behaviors.

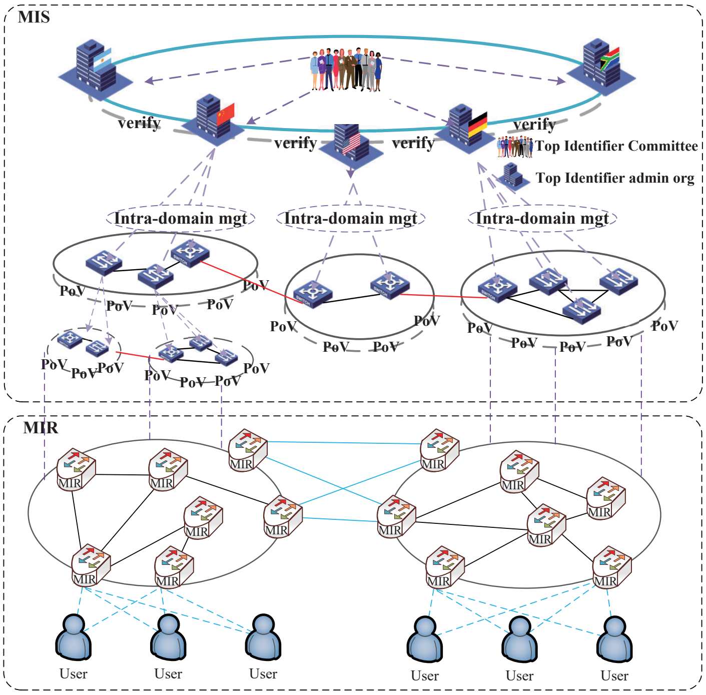


Figure 4. Separation of multi-identifier network management plane and data plane.

3.2. Encryption and Blockchain Protection

Identity authentication is based on Elliptic Curve Cryptography (ECC), rather than RSA. In addition to the registration phase, the subsequent authentication phase does not require the participation of a trusted third party, which ensures communication security and reduces the overhead of computing and communication. Data transmitted in the network will be protected by hash and asymmetric encryption at the packet level, rather than at the channel level, to prevent sensitive information from being eavesdropped, intercepted, or tampered with.

The nodes in the MIS consortium chain are divided into committee nodes, accounting nodes, and ordinary user nodes. The committee nodes are elected by members who volunteer to maintain the consortium chain and have equal voting rights jointly. The bookkeeping node is a node with the privilege to produce blocks voted by the committee nodes.

During user registration, the user will generate his public and private key pair, package the public key and the real identity information signed by the private key, and submit it to any blockchain node in the MIS system. The node will check the format of the registration request and find whether the user information already exists in the local database to avoid repeated registration. After that, this node will perform primary verification of the content of the registration information, package it into blockchain transaction information and submit it to the accounting node in the MIS consortium blockchain, and store the transaction information in the transaction pool.

We adopted the PoV consensus algorithm [33] to accelerate the process of consortium blockchain consensus, as depicted in Figure 5. At the beginning of the blockchain consensus, the accounting node will take out some transaction information from the transaction pool to generate a pre-block, and send it to the committee node in the consortium chain to request a signature. The committee node verifies the pre-block header and the content of each transaction, and sends it back to the accounting node after signing. If the billing node receives more than half of the signatures of the committee nodes within a given time threshold, it will store the signature information in the block header and set the timestamp to write this pre-block to the master of the consortium chain, and the pre-block becomes an official block. Otherwise, the pre-block will be deleted; the transaction will be taken out from the transaction pool again, and the pre-block will be generated and sent to all committee nodes for verification and signature.

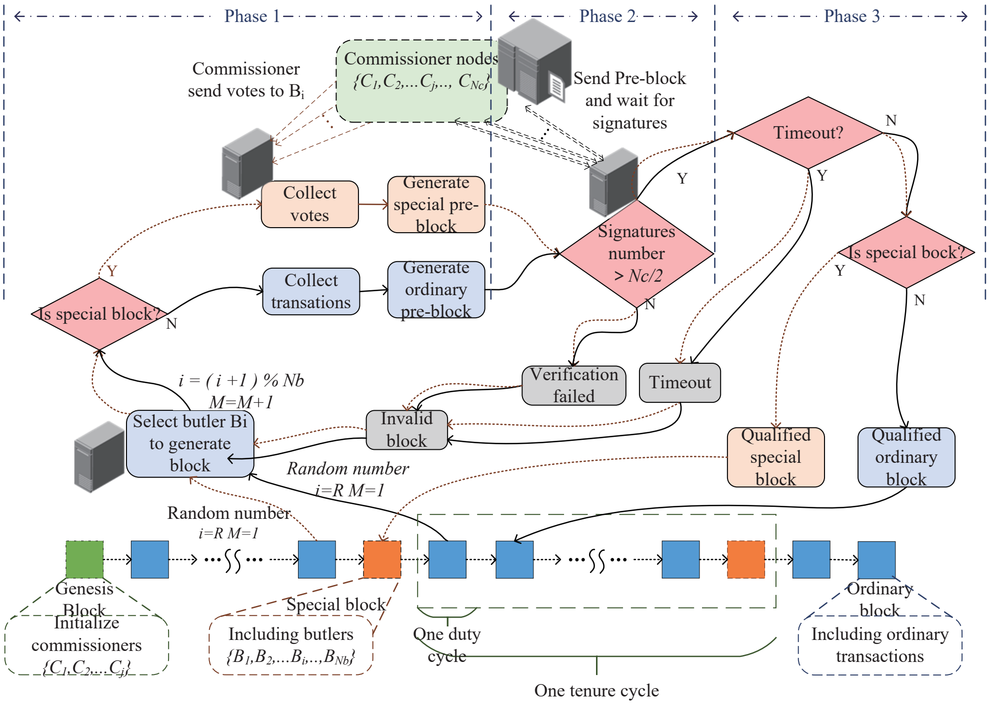

Phase 1
Phase 2
Phase 3
Commissioner send votes to B$_i$
Commissioner nodes {C_1,C_2,...C_j,.., C_{Nc}}
Send Pre-block and wait for signatures
Collect votes
Generate special pre-block
Timeout?
N
Y
Y
Signatures number > $Nc/2$
Is special bock?
Y
Is special block?
N
Collect transations
Generate ordinary pre-block
N
N
$i = (i +1) \% Nb$
$M=M+1$
Verification failed
Timeout
Select butler Bi to generate block
Invalid block
Qualified special block
Qualified ordinary block
Random number $i=R$ $M=1$
Random number $i=R$ $M=1$
Genesis Block
Initialize commissioners {C_1,C_2,...C_j}
Special block
Including butlers {B_1,B_2,...B_i,..,B_{Nb}}
One duty cycle
One tenure cycle
Ordinary block
Including ordinary transactions

Figure 5. PoV consensus and block generation procedure.

When a user performs network data transmission, the user's signature information will be included in the data packet to record the user's access path in the multi-identifier network and realize the traceability of network security. When abnormal access is detected, the registered identity information reserved by the user will be retrieved from the consortium chain database, and the user will be warned or prohibited.

3.3. Trusted Computing Whitelist

Trustworthiness is the expectation that an entity can acquire the expected effect when realizing a given goal. It should be salable, adaptive, and auto-configurable, especially for the ubiquitous computing scenarios, such as Internet of Things and Internet of Vehicles [34]. We adopted the active immune trusted computing module (TCM) independently developed by China [13] in the client host hardware to save the encryption keys used in the consensus process of the blockchain nodes.

The private key is stored in the TCM chip and cannot be leaked and read. The public key can be used everywhere to identify the node identity and for signature verification. At the same time, all kinds of data generated by the node (such as blocks, transactions) are signed using the signature algorithm provided by TCM, to protect the private data such as encryption keys, and ensure data integrity, confidentiality, and security of the data through the transmission of the trusted chain. Identity authentication and encryption key mechanisms ensure the credibility of the computing environment.

Whitelist is adopted for application identification, and protection, such as NFD, PSYNC; only a few trusted applications are allowed to run in the current network system to realize the supervision and protection of the whole life cycle of applications from startup, loading, and operation, which reduces the load of the system and improves the availability of the system, as shown in Figure 6. The application developers use their private key to issue the application signatures and register them with the MIS, or the MIS trusted service to provide the verification benchmark and establish the application white list database. For applications on the whitelist, we conduct essential behavior analysis on their regular operations, and establish the whitelist behavior rule base of the applications. When an application starts, the hash measurement value of the application is obtained and compared with the expected benchmark value taken from the trusted module (TCM) to complete the integrity measurement. The execution of the application will be allowed only when the application is integrity and has not been tampered with. At the same time, during the operation of the application, the running status of its key behaviors is monitored in real-time, and the pre-established application behavior rule base is compared to detect abnormal behaviors in time, and record their operations in the log, to trade off between security and availability, and ensure the orderly operation of the application.

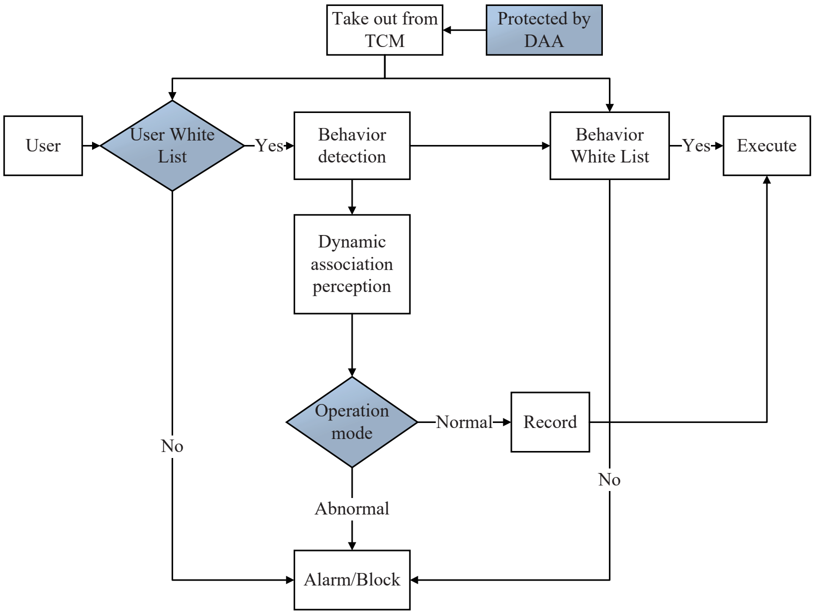


Figure 6. Trusted computing remote attestation and whitelisting.

At the same time, the security of TCMs needs to be guaranteed. [35] The feasibility of trusted computing storage and computing environment can be confirmed remotely through remote attestation Direct Anonymous Attestation (DAA) technology [14]. DAA uses the Carmenisch–Lysyanskaya signature mechanism [36] to sign certificates of the public key generated by the TCM to ensure the legitimacy of the TCM module. The TCM module uses the DAA certificate to interact with the DAA verifier, thereby identifying and distinguishing the compromised TCM module and ensuring the security of the trusted root.

4. Dynamic Defense

Static defense methods, such as identity authentication, data signatures, blockchain, and trusted computing, provide sufficient external barriers for network security to obtain balanced network confidentiality, integrity, and availability and hence protecting network nodes' security. They can effectively prevent and trace security problems. At the same time, nodes in the network are not simply isolated. They may face group network problems and new types of severe or unknown attacks. Therefore, it is necessary to build an effective dynamic defense system. The measures taken into consideration by DDESS include situation awareness and network security vaccine training and distribution.

4.1. Situation Awareness

Network situation awareness refers to the acquisition, understanding, and display of security elements that can bring about the network situation changes in a large-scale network environment, as well as the prediction of network development trends [37,38]. It is divided into network security situational extraction, understanding, and prediction.

When a network individual conducts a static defense, numerous original log files and backbone network traffic data will be generated, including user usage logs and alarm information, which provides a wealth of training materials for dynamic defense situation awareness of network security.

DDESS builds a decision tree containing all or part of the rules in the security policy rule set on the log file, which parses, filters, omissions, and normalizes the entries in the log file. Firstly, the raw data generated by the network monitoring equipment and management system are preprocessed, including data cleaning, noise reduction, dimensionality reduction, standardized merging or conversion, and data verification, removing duplicate and redundant information, merging similar information, and correcting error information, in order to obtain standardized asset data sets, threat data sets, and vulnerability data sets. Then, it carries out data correlation analysis and uses expert knowledge to model network activities and their regulations and characteristics, and identifies the existence and forms of various individuals in the network, and further identifies three different behaviors: malicious attack behavior, abnormal risk behavior (including weak password, account risk login, remote control), and normal access behavior. Behaviors that do not conform to the behavior baseline will trigger behavior alarms to detect risks in time and better find hidden attacks.

Based on the identified attack activities and their characteristics, the attacker's intention is inferred by further analyzing the semantics of these attack activities and the possible correlation among them. Its main tasks include identifying the source and type of these attack activities and judging the attacker's capability, opportunity, and the possibility of conducting a successful attack. Based on the network attack chain, network traffic abnormalities are found, such as external attacks, internal malicious scanning, ARP spoofing attack, and internal illegal accesses. The attacker's intentions can be mainly analyzed from the attack's behavior and target. The attack behavior prediction analyzes the logical relationship between attack behaviors and infers possible changes. In addition, we need to consider the function and importance of network assets to infer the attacker's attack intention and source. DDESS situational understanding draws portraits of user behaviors, including individuals and groups. Moreover, it constructs the network attack knowledge graph and draws the security graph from different perspectives, such as external attacks, internal horizontal penetration, and network data leakage. Therefore, DDESS can understand the current overall network security situation, detect and discover security events, and analyze and evaluate the network vulnerabilities and the impact of attacks.

DDESS first checks the network's security status to achieve network security situation awareness. Then, it understands in detail the various assets and participants within the networks and their potential vulnerabilities to attacks. It finally draws a graph of the network assets and the corresponding security vulnerabilities. After comprehensively acquiring network threat status data, convolutional neural networks (CNN) [39] are used to evaluate the potential network security risks and the behavioral patterns of different nodes. The resulting infliction of existing attack behaviors is based on the current network status and the identified attack activities, along with the vulnerabilities of network assets, as shown in Algorithm 1.

Algorithm 1 Training of security situational project model

```
Input:
    The total number of layers $L$, the number of neurons in each hidden layer and output layer,
    activation function $f$, loss function, iteration step $\eta$,
    maximum iteration times $MAX$ and stop iteration threshold $\epsilon$
Output:
  linear relation coefficient matrix $\omega$ and bias vector $b$ of each hidden layer and output layer.
Input the risk vector dataset $R = \{r_1, r_2, r_3, \cdots, r_n\}$.
for m=1 to L do
    Extract the $m-$layer features $\delta^m$ with a convolution kernel
    $\delta^m(r) \leftarrow f(\sum \omega^m \delta^{m-1} + b_m)$
    Reduce network scale with max pooling:
    $\delta^m(r) = maxpooling(o^{m-1}(r) + b_m)$
end for
update the weight $\omega$ with backpropagation:
for $l = L$ to 2 do
    Compute $\delta^l$ based on $\delta^{l+1}$ and $\omega^{l+1}$ and $z^l$
    Compute the gradient $\bigtriangledown\omega^{l+1}$ and $\bigtriangledown b^l$
    Update $\omega^l$ and $b^l$ in $l$th layer:
    $\omega^l \leftarrow \omega^l - \eta \sum_{i=1}^{m} \delta^{i,l} (a^{i,l-1})^T$
    $b^l \leftarrow b^l - \eta \sum_{i=1}^{m} \delta^{i,l}$
    if $\Delta\omega < \epsilon$ or $L > MAX$ then
        break;
    end if
end for
```

The overall time complexity of Algorithm 1 is the cumulative time complexity of all convolution layers

$$\text{Time} \; : O\left(\sum_{l=1}^{D} m_l^2 \cdot f_l^2 \cdot C_{l-1} \cdot C_l\right), \tag{1}$$

and the spatial complexity is

$$\text{Space} \; : O\left(\sum_{l=1}^{D} f_l^2 \cdot C_{l-1} \cdot C_l + \sum_{l=1}^{D} m_l^2 \cdot C_l\right) \tag{2}$$

where D is the network depth, f_l is the filter size, C_l is the filter number and the output channels number of layer l, and C_{l-1} also represents the input channels number of layer l. m_l is the size of the output feature map, and $m_l = \lfloor (n_l - f_l + 2 * p_l)/s_l \rfloor + 1$, where n_l is the size of the input matrix, p_l is the padding, and s_l is the stride. In addition, we eliminate gradient vanishing problem with batch normalization [40] and introduce max pooling layer and 1×1 convolution kernel [41] for dimensionality reduction to reduce both the time and space complexity. DDESS can predict the future security status and the changing trend of the network and takes effective defensive measures.

4.2. Network Security Vaccine Training and Herd Immunity

The active immune trusted computing technology adopted in static defense can provide immune capabilities for network information systems. The vaccine culture used in organisms usually selects pathogens with strong immunogenicity, loses toxicity through biochemical inactivation or repeatedly immune iteration of animal cells, and retains its immunogenicity.

Biological vaccines are usually injected into organisms to produce immunity. However, network vaccines are different from biological vaccines in a way that they aim to provide a real-time and automatic defense. While defending against attacks, such vaccines improve the network's resilience and elasticity to maintain its availability and thus return to a normal state eventually. Therefore, the cultivation of network defense vaccines needs to be carried out dynamically in the network environment. The network vaccine can be described as:

$$V : (antigen, srcaddr, destaddr, timestamp, protocol). \tag{3}$$

The vaccine of network security's immunity is a string extracted from the characteristics of data packets transmitted over the network. Antibodies are the measures taken in static defenses and the rules generated in dynamic defenses. During the matching process of antigens and antibodies, the immune cells constantly defend against network attacks, form clonal regeneration, and evolve into memory cells. The memory cells record accurately the network intrusions that have occurred, and encapsulate them into a vaccine cell after stamping them with a timestamp according to the vaccine format as shown in Equation (3).

The encapsulated network security vaccine is transmitted to the vaccine defense center of the adjacent network to be finally distributed among the network nodes. The selection of such a network considers the local routing table, routing, and forwarding. When the antigenic match with the vaccine happens, it becomes activated and can detect network attacks. At the same time, network nodes can also be used for training and cultivating network vaccines as described by Equation (3). When new types of attacks are discovered, these nodes will continue training and learning to enhance the vitality of antibodies in vaccine cells to improve the autoimmunity and achieve adversarial collaboration.

5. Experiments

We evaluated the overall defense effectiveness, and intrusion detection performance of DDESS compared with related state-of-the-art schemes in this section.

5.1. Overall Performance Evaluation

Experiments were conducted on a testbed using the topology presented in Figure 7. The router *pkusz*11 plays the role of the edge router that is connected to the Internet. Experiments and simulations are based on the published KDD-99 dataset that consists of 5 million records [42]. These records are totally made up of 41 attributes and one attack category field which marks all observations as either "normal" or "attacked" with one of the following attacks: Denial of Service (DoS), Remote to Local (R2L), User to Root (U2R), and Probing/surveillance. We carry out attack experiments on DDESS combined with the multi-identifier network. The selected attack methods are conventional under IP networks, such as target detection, attack injection, ARP attacks, and so on. Experiments' results are demonstrated in Table 1, where ✓ indicates that the attack was successful, while ✗ represents an attack failure.

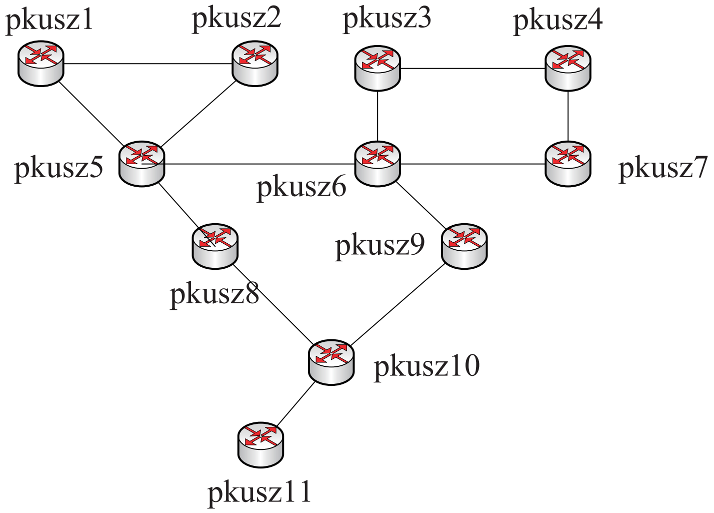


Figure 7. DDESS performance evaluation topology.

Table 1. Attack results of experiments.

Attack Phase	Description		IP-IP	IP-MIN
Target detection	Host discovery		✓	✗
	ping scan		✓	✗
	OS recognition		System fingerprints obtained	Host non-survival
	Port scan		All ports probed	The host is alive, but no port detected
Attack injection	Trojan	TCP trojan	✓	✗
		UDP trojan	✓	✗
		ICMP trojan	✓	✗
	One-sentence shell		✓	✗
Action	ARP Attack		Information sniffing	Target cannot be sniffed
			Network disconnection attack	Target not affected

MIN can effectively defend against target detection attacks such as host discovery, ping scanning, and operating system recognition. It can also prevent Trojan attack injections and one-sentence shell attacks. Attacks within the action phase were divided into ARP disconnection and ARP spoofing. We used "arpspoof" to send fake MAC-IP binding packets, hence poisoning the gateway's ARP cache. The attacker can therefore disconnect the target network or monitor its traffic. For the ARP disconnection attack, ARP spoofing was first initiated with an arpspoof tool to change the IP forwarding path from one target host to another within the LAN. Both types of ARP attacks were successfully blocked in MIN as shown in Table 1.

5.2. Performance of Dynamic Defense

We compared the intrusion detection performance of DDESS on the KDD-99 dataset [42] with that of state-of-the-art machine learning based methods with 10-fold cross-validation. Our comparison references included Gradient Boosted Machine (GBM) [43], k-Nearest Neighbor (kNN) [44], Classification and Regression Trees (CART) [45], Multi-Layer Perceptron (MLP) [46], and AdaBoost [47]. The evaluation results are depicted in Figure 8.

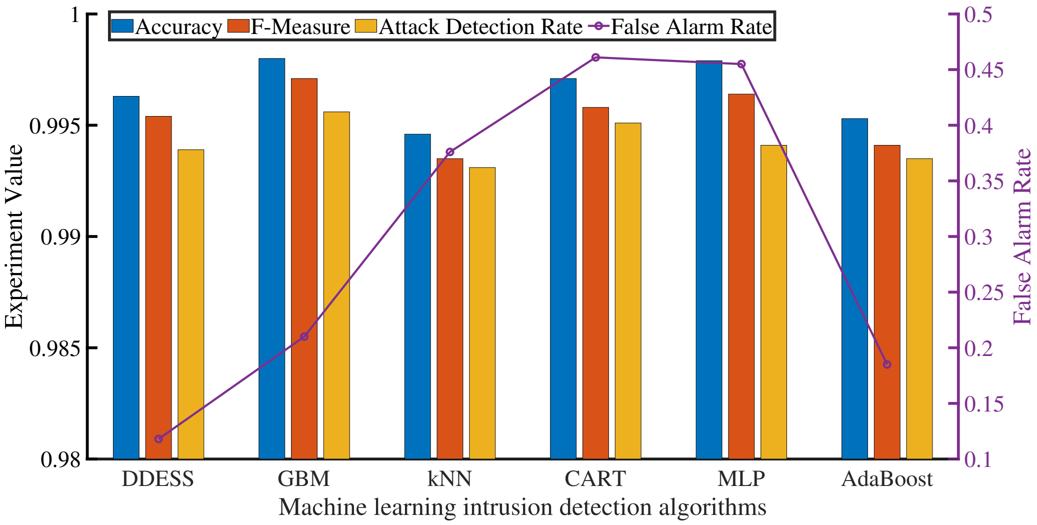


Figure 8. Performance comparison of intrusion detection.

It was observed that, compared with other algorithms, GBM, MLP, and CHAT have better performance in accuracy, reaching 99.81%, 99.79%, and 99.71%, respectively. In order to trade-off the precision rate and recall rate, F-Measure was usually applied [48],

$$\text{F-Measure} = \frac{(1+\beta^2)P \times R}{\beta^2 P + R}, \tag{4}$$

where P means the precision rate and R represents the recall rate. Here, we chose $\beta^2 = 1$ because we attached importance of both P and R. GBM still achieved the best F-Measure value (99.71%) across all the algorithms, whereas kNN obtains the worst one (99.35%).

As far as attack detection rate (ADR) is concerned, algorithms which are adaptive to continuous attacks have better effects. In Figure 8, GBM (99.56%) and CART (99.51%) can outperform other methods in ADR.

In addition, DDESS can acquire the best false alarm rate (0.118%) because the deep neural networks can reduce the opportunities for false positives. A low false alarm rate and low latency are important for time-sensitive networks (TSN) of Internet of Things (IoT) [31]. Therefore, in order to better monitor network security, we proposed DDESS to comprehensively measure these indicators and come to trade-offs.

We further evaluated the prediction performance of dynamic defense and compared it with TSA-AdaBoost [49], a situation awareness algorithm based on the AdaBoost machine learning method. DDESS adopts the batch normalization [40] to reduce the influence of gradient vanishing problem, and max-pooling layer and 1 × 1 convolutional kernel [41] for dimensionality reduction. Therefore, the overall fitness of the proposed DDESS is better than TSA-AdaBoost for the prediction of test samples, as shown in Figure 9a. We select mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE) as the prediction evaluation metrics to evaluate the proposed prediction model, where

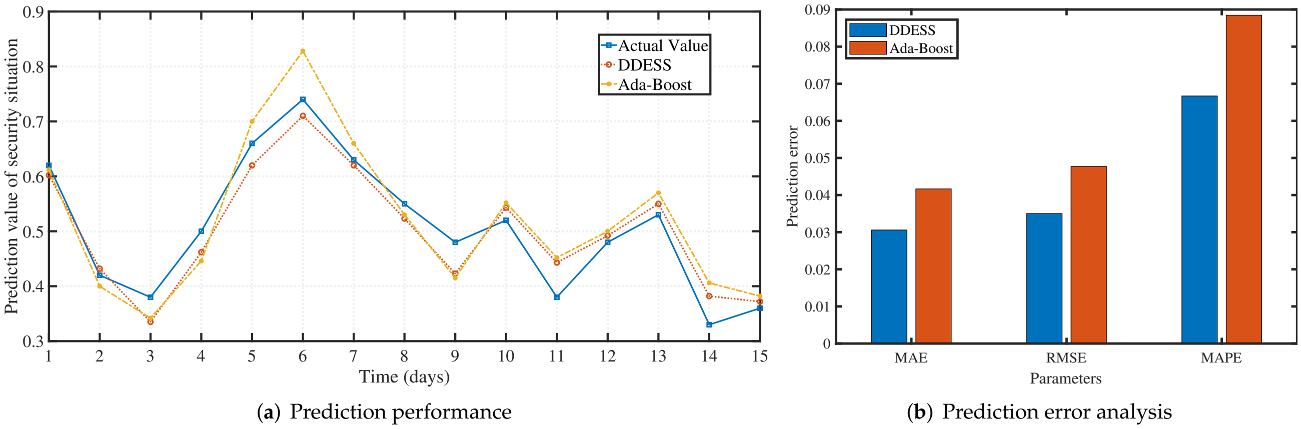
$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} |\hat{y}_i - y_i| \tag{5}$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (\hat{y}_i - y_i)^2} \tag{6}$$

$$\text{MAPE} = \frac{100\%}{N}\sum_{i=1}^{N} \left| \frac{\hat{y}_i - y_i}{y_i} \right|, \tag{7}$$

where $\hat{y}_i$ means the prediction situation value. As shown in Figure 9b, the MAE and RMSE values of DDESS are 0.0306 and 0.035, respectively, while these of TSA-AdaBoost are 0.0417 and 0.0486, respectively. Compared with TSA AdaBoost, DDESS has less overall error precision in predicting network security situation value. In addition, the MAPE value of DDESS and TSA-AdaBoost is 6.67% and 8.95%, respectively, which means that DDESS has better accuracy than TSA AdaBoost.

Let us analyze forecast results in Figure 9a from a network administrators' perspective. Although the situational value on day 3 is relatively low, it is predicted that the network situational value will have a higher tendency on the next day (i.e., day 4), which will warn network administrators about the occurrence of network attacks. At the same time, based on the 11th-day forecast trend, this is likely to be the end phase of network attacks, which will make administrators pay special attention to the network behavior logs for the next two days to ensure that they are not deleted or destroyed by attackers. On the other hand, the forecasts on the 12th and 13th show that network attacks continue, and some vulnerabilities probably exist in the network system.



(**a**) Prediction performance (**b**) Prediction error analysis

Figure 9. Prediction performance comparison on situation awareness.

DDESS can train the attack model and make corresponding portraits of the attack and the attacker without requiring additional hardware, such as FPGA. To perceive the motivation behind the attack, DDESS will classify and mark the attack behaviors of the same source. This helps learn and predict potential attacks behaviors from the same source in the future.

5.3. Competition and Trial

The multi-identifier network system (MIN) combined with DDESS technology has withstood the "Network Security Challenge Competition" held by the Purple Mountain Laboratory [50]. Forty-eight teams continuously carried out remote online high-intensity network attacks for 72 h. During this period, the cumulative number of attacks reached 3.58 million. More precisely, the MIN was one of the few network security systems any team has not broken through. It also successfully prevented most competition-related qualification attacks such as Linux privilege escalation, virtual machine escape, session forging and hijacking, and simulation PWN. The DDESS-based system has been proved to be more robust when compared with traditional commercial workarounds.

6. Conclusions and Future Work

To overcome the current internet security problems, we propose a double defense strategy with endogenous safety and security (DDESS) based on the MIN system. DDESS provides both static and dynamic defense strategies against different network attacks. The proposed system preserves network security using static defensive measures such as identity verification, encryption protection, and trusted computing. It also uses dynamic defensive measures like situation awareness and vaccine cultivation and distribution. Experiments and performance analysis showed that DDESS provides sufficient availability and robust security compared to other existing network defense solutions. In the future work, we plan to introduce edge computing technology to reduce the pressure of central servers and better adapt to the network development, such as 6G, Internet of things (IoT), etc. In addition, we will study the intrusion detection algorithms, and put forward more effective strategies for network security detection and prediction in our future work.

Author Contributions: Conceptualization, H.L. and Y.W.; methodology, H.Z.; software, Y.W.; validation, A.S. and Y.W.; formal analysis, Y.W.; investigation, Y.W.; resources, H.L.; data curation, H.Z.; writing—original draft preparation, Y.W.; writing—review and editing, A.S.; visualization, A.S.; supervision, A.S.; project administration, H.L.; funding acquisition, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Guangdong R&D Key Program (2019B010137001), Shenzhen Research Program (GXWD20201231165807007-20200807164903001, JCYJ20210324122013036, JCYJ20190808155607340), China Environment for Network Innovation (CENI) (GJFGW No.[2020]386, SZFGW [2019] 261), and the National Keystone R&D Program of China (2017YFB0803204).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used for this research is publicly available at https://kdd.org/kdd-cup/view/kdd-cup-1999/Data.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DDESS	Double Defense strategy with Endogenous Safety and Security
NDN	Named Data Networking
MIN	Multi-Identifier Network
MIS	Multi-Identifier System
MIR	Multi-Identifier Router
PoV	Proof of Vote
DAA	Direct Anonymous Attestation
TCM	Trusted Computing Module
ACC	Accuracy
ADR	Attack Detection Rate
FAR	False Alarm Rate

References

1. Blumenthal, M.S.; Clark, D.D. Rethinking the design of the internet: The end-to-end arguments vs. the brave new world. *ACM Trans. Internet Technol.* **2001**, *1*, 70–109. [CrossRef]
2. Kouicem, D.E.; Bouabdallah, A.; Lakhlef, H. Internet of things security: A top-down survey. *Comput. Netw.* **2018**, *141*, 199–221. [CrossRef]
3. Contreras-Castillo, J.; Zeadally, S.; Guerrero-Ibañez, J.A. Internet of vehicles: Architecture, protocols, and security. *IEEE Internet Things J.* **2017**, *5*, 3701–3709. [CrossRef]
4. Park, S.M.; Kim, Y.G. A Metaverse: Taxonomy, components, applications, and open challenges. *IEEE Access* **2022**, *10*, 4209–4251. [CrossRef]
5. Wu, J. *Cyberspace Endogenous Safety and Security: Mimic Defense and Generalized Robust Control*; Science Press: Beijing, China, 2020.
6. Forrest, S.; Hofmeyr, S.A.; Somayaji, A. Computer immunology. *Commun. ACM* **1997**, *40*, 88–96. [CrossRef]
7. Dasgupta, D. Advances in artificial immune systems. *IEEE Comput. Intell. Mag.* **2006**, *1*, 40–49. [CrossRef]
8. Yu, Q.; Ren, J.; Zhang, J.; Liu, S.; Fu, Y.; Li, Y.; Ma, L.; Jing, J.; Zhang, W. An Immunology-Inspired Network Security Architecture. *IEEE Wirel. Commun.* **2020**, *27*, 168–173. [CrossRef]
9. Li, H.; Yang, X. Architecture of Sovereignty Network. In *Co-Governed Sovereignty Network: Legal Basis and Its Prototype & Applications with MIN Architecture*; Springer: Singapore, 2021; pp. 61–94. [CrossRef]
10. Rose, S.W.; Borchert, O.; Mitchell, S.; Connelly, S. Zero Trust Architecture, Special Publication (NIST SP). 2020. Available online: https://www.nist.gov/publications/zero-trust-architecture (accessed on 29 November 2021).
11. Wang, Y.; Li, H.; Huang, T.; Zhang, X.; Bai, Y. Scalable Identifier System for Industrial Internet Based on Multi-Identifier Network Architecture. *IEEE Internet Things J.* **2021**, 1. [CrossRef]
12. Li, K.; Li, H.; Hou, H.; Li, K.; Chen, Y. Proof of vote: A high-performance consensus protocol based on vote mechanism & consortium blockchain. In Proceedings of the IEEE HPCC, Bangkok, Thailand, 18–20 December 2017.
13. Shen, C. To Create a Positive Cyberspace by Safeguarding Network Security with Active Immune Trusted Computing 3.0. *J. Inf. Secur. Res.* **2018**, *4*, 282–302.
14. Brickell, E.; Camenisch, J.; Chen, L. Direct anonymous attestation. In Proceedings of the 11th ACM conference on Computer and communications security, Washington, DC, USA, 25–28 October 2004; pp. 132–145.
15. Kaur, T.; Malhotra, V.; Singh, D. Comparison of network security tools-firewall, intrusion detection system and Honeypot. *Int. J. Enhanc. Res. Sci. Technol. Eng.* **2014**, *3*, 200–204.
16. Song, X. Firewall technology in computer network security in 5G environment. *J. Phys. Conf. Ser.* **2020**, *1544*, 012090. [CrossRef]
17. Aman, M.N.; Basheer, M.H.; Sikdar, B. Data provenance for IoT with light weight authentication and privacy preservation. *IEEE Internet Things J.* **2019**, *6*, 10441–10457. [CrossRef]
18. Sandhu, R.S.; Samarati, P. Access control: Principle and practice. *IEEE Commun. Mag.* **1994**, *32*, 40–48. [CrossRef]

19. Tidjon, L.N.; Frappier, M.; Mammar, A. Intrusion detection systems: A cross-domain overview. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 3639–3681. [CrossRef]
20. Williams, L.; McGraw, G.; Migues, S. Engineering security vulnerability prevention, detection, and response. *IEEE Softw.* **2018**, *35*, 76–80. [CrossRef]
21. Kausar, F.; Aljumah, S.; Alzaydi, S.; Alroba, R. Traffic analysis attack for identifying users' online activities. *IT Prof.* **2019**, *21*, 50–57. [CrossRef]
22. Roger, M.; Goubault-Larrecq, J. Log auditing through model-checking. In Proceedings of the Proceedings of the 14th IEEE workshop on Computer Security Foundations, Cape Breton, NS, Canada, 11–13 June 2001; pp. 220–234.
23. Sun, Y.; Tian, Z.; Li, M.; Su, S.; Du, X.; Guizani, M. Honeypot Identification in Softwarized Industrial Cyber–Physical Systems. *IEEE Trans. Ind. Inform.* **2020**, *17*, 5542–5551. [CrossRef]
24. Fan, W.; Du, Z.; Fernández, D.; Villagrá, V.A. Enabling an anatomic view to investigate honeypot systems: A survey. *IEEE Syst. J.* **2017**, *12*, 3906–3919. [CrossRef]
25. Shrivastava, R.K.; Bashir, B.; Hota, C. Attack detection and forensics using honeypot in IoT environment. In Proceedings of the International Conference on Distributed Computing and Internet Technology, Bhubaneswar, India, 10–13 January 2019; pp. 402–409.
26. Zhuang, R.; DeLoach, S.A.; Ou, X. Towards a theory of moving target defense. In Proceedings of the ACM Workshop on Moving Target Defense, Scottsdale, AZ, USA, 7 November 2014; pp. 31–40.
27. Wu, J. *Cyberspace Mimic Defense*; Springer: Cham, Switzerland, 2020.
28. Wang, C.; Lu, Z. Cyber deception: Overview and the road ahead. *IEEE Secur. Priv.* **2018**, *16*, 80–85. [CrossRef]
29. Makarfi, A.U.; Rabie, K.M.; Kaiwartya, O.; Adhikari, K.; Nauryzbayev, G.; Li, X.; Kharel, R. Toward Physical-Layer Security for Internet of Vehicles: Interference-Aware Modeling. *IEEE Internet Things J.* **2020**, *8*, 443–457. [CrossRef]
30. Kumar, S.; Singh, K.; Kumar, S.; Kaiwartya, O.; Cao, Y.; Zhou, H. Delimitated anti jammer scheme for Internet of vehicle: Machine learning based security approach. *IEEE Access* **2019**, *7*, 113311–113323. [CrossRef]
31. Wu, J. Thoughts on the development of novel network technology. *Sci. China Inf. Sci.* **2018**, *61*, 101301. [CrossRef]
32. Zhang, L.; Afanasyev, A.; Burke, J.; Jacobson, V.; Claffy, K.; Crowley, P.; Papadopoulos, C.; Wang, L.; Zhang, B. Named data networking. *ACM SIGCOMM Comp. Commun. Rev.* **2014**, *44*, 66–73. [CrossRef]
33. Li, H.; Yang, X. Key Technologies of Sovereignty Network. In *Co-Governed Sovereignty Network*; Springer: Cham, Switzerland, 2021; pp. 95–182.
34. López, J.; Maña, A.; Muñoz, A. A secure and auto-configurable environment for mobile agents in ubiquitous computing scenarios. In Proceedings of the International Conference on Ubiquitous Intelligence and Computing, Wuhan, China, 3–6 September 2006; pp. 977–987.
35. Gürgens, S.; Rudolph, C.; Scheuermann, D.; Atts, M.; Plaga, R. Security evaluation of scenarios based on the TCG's TPM specification. In Proceedings of the European Symposium on Research in Computer Security, Oslo, Norway, 11–13 September 2007; pp. 438–453.
36. Camenisch, J.; Lysyanskaya, A. A signature scheme with efficient protocols. In Proceedings of the International Conference on Security in Communication Networks, Amalfi, Italy, 11–13 September 2002; pp. 268–289.
37. Endsley, M.R. Design and evaluation for situation awareness enhancement. In Proceedings of the Human Factors Society Annual Meeting, Anaheim, CA, USA, 24 October 1988; Volume 32, pp. 97–101.
38. Bass, T. Intrusion detection systems and multisensor data fusion. *Commun. ACM* **2000**, *43*, 99–105. [CrossRef]
39. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
40. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 448–456.
41. Lin, M.; Chen, Q.; Yan, S. Network in network. *arXiv* **2013**, arXiv:1312.4400.
42. Tavallaee, M.; Bagheri, E.; Lu, W.; Ghorbani, A.A. A detailed analysis of the KDD CUP 99 data set. In Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, Canada, 8–10 July 2009; pp. 1–6.
43. Tama, B.A.; Rhee, K.H. An in-depth experimental study of anomaly detection using gradient boosted machine. *Neural Comput. Appl.* **2019**, *31*, 955–965. [CrossRef]
44. Li, W.; Yi, P.; Wu, Y.; Pan, L.; Li, J. A new intrusion detection system based on KNN classification algorithm in wireless sensor network. *J. Electr. Comput. Eng.* **2014**, *2014*, 240217. [CrossRef]
45. Soylu, T.; Erdem, O.; Carus, A.; Güner, E.S. Simple CART based real-time traffic classification engine on FPGAs. In Proceedings of the 2017 International Conference on ReConFigurable Computing and FPGAs (ReConFig) (ReConFig), Cancun, Mexico, 4–6 December 2017; pp. 1–8.
46. Ahmad, I.; Abdullah, A.; Alghamdi, A.; Alnfajan, K.; Hussain, M. Intrusion detection using feature subset selection based on MLP. *Sci. Res. Essays* **2011**, *6*, 6804–6810.
47. Hu, W.; Hu, W. Network-based intrusion detection using Adaboost algorithm. In Proceedings of the The 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05), Compiegne, France, 19–22 September 2005; pp. 712–717.
48. Fawcett, T. An introduction to ROC analysis. *Pattern Recognit. Lett.* **2006**, *27*, 861–874. [CrossRef]

49. Zhao, Y.; Cheng, G.; Duan, Y.; Gu, Z.; Zhou, Y.; Tang, L. Secure IoT edge: Threat situation awareness based on network traffic. *Comput. Netw.* **2021**, *201*, 108525. [CrossRef]
50. Laboratories, P.M. Mimic Defense International Elite Challenge. 2021. Available online: http://www.pmlabs.com.cn/plus/view.php?aid=1464 (accessed on 29 November 2021).