# A Trust Management Method Against Abnormal Behavior of Industrial Control Networks Under Active Defense Architecture

Jingpei Wang, Zhenyong Zhang, and Mufeng Wang

***Abstract*—Trusted computing is a typical active defense technology. Trust management is a core support technology of trusted computing. However, when trust management is applied in the industrial control systems, how to identify malicious behavior effectively, model trust relationships, and make a decision based on behavior trustworthiness, meanwhile how to ensure deployed trust mechanism does not affect the control network's availability, is a significant issue that has not been solved in the previous literature. This paper proposes a trust management method against abnormal behavior of industrial control networks under active defense architecture. Firstly, we review the difficulties of trust management when applied to industrial control networks and analyze abnormal behaviors of the control operations under unknown threats. Then we extract trust information, model the trust relationship of abnormal behaviors, and establish a trust update and decision-making mechanism under the availability constraints of industrial control networks. Furthermore, we provide a deployment method of the proposed trust management in a distributed control network. Finally, we take five typical abnormal operations on control instruction in an industrial control network as an example and perform a detailed analysis and experimental verification of the proposed method. The results prove that the proposed trust management method has good immunity to abnormal behaviors of the control flow and can be deployed in an industrial control system with availability constraints.**

***Index Terms*—Industrial control networks, trust management, availability, trustworthy.**

## I. Introduction

INDUSTRIAL control networks are widely used in critical industrial areas, i.e., nuclear infrastructures, petroleum and petrochemical industry, water conservancy, natural gas, and advanced manufacturing, playing the role of the central nervous system [1]. Due to inherent security vulnerabilities, the tendency of the open interconnection of physical information systems, the widespread application of general intelligent components into the industrial control networks, security threats, such as viruses, Trojans, are spreading in industrial control networks. A series of famous industrial incidents, i.e., "Stuxnet," "Duqu," "flames," "dragonfly," and "Ukrainian blackouts," have occurred in succession. Industrial network attacks seriously threaten the regular operation of critical infrastructures, the security protection of industrial control networks has become a hot spot [2]–[3]. Some researchers are concerned about security protection policies [4]–[5]. Some typical security protection technologies have been proposed, including vulnerability identification of the SCADA (Supervisory Control and Data Acquisition) system [6], security protection of industrial terminals [7], and privacy protection of industrial networks [8], etc.

However, the security protection of industrial control networks is difficult. On one hand, the industrial control systems have some characteristics: 1) high availability, availability > integrity > confidentiality, which is the opposite of the ordinary IT system; 2) high requirements of continuity and reliability, it needs real-time communication, and no restart is allowed. It emphasizes system reliability and operational compliance; 3) Industrial field network includes a wide variety of proprietary devices, which are developed and operated mostly based on embedded systems (for example, VxWorks, WinCE, etc.) and use proprietary communication protocols (such as OLE for Process Control (OPC), Modbus, Distributed Network Protocol 3 (DNP3), etc.). However, the information security methods in IT information systems mostly adopt encryption and access control, general secure routing and protocols, and updatable operating systems, which cannot be directly applied to industrial networks. Moreover, the traditional Firewall, Intrusion Detection System (IDS), and Intrusion Prevention System (IPS) may not identify the industrial protocols or block important protocols then interfering with the availability [4], [9]. On the other hand, due to the advancement of vulnerability mining and network attack technology, the protection technology lags. 1) Due to the limitations of human cognition, the designed IT system can not avoid unknown threats. The anti-virus software, firewall, and intrusion prevention system generally identify known threats through the threat pattern library. Although some advanced IPS have prediction functions, it is still difficult to resist unknown

Manuscript received 25 November 2020; revised 28 July 2021 and 29 December 2021; accepted 2 May 2022. Date of publication 9 May 2022; date of current version 12 October 2022. This work was supported by the National Nature Science Foundation of China (61972345, U1911401), and was partly supported by the Zhejiang Provincial Nature Science Foundation of China (LZ21F030004). The associate editor coordinating the review of this article and approving it for publication was M. Tornatore. *(Corresponding author: Mufeng Wang.)*

Jingpei Wang is with the College of Control Science and Engineering, Zhejiang University, Hangzhou 310063, China (e-mail: wjpbupt@163.com).

Zhenyong Zhang is with the College of Computer Science and Technology, Guizhou University, Guiyang 550025, China (e-mail: zyzhangnew@gmail.com).

Mufeng Wang is with the China Industrial Control Systems Cyber Emergency Response Team, Beijing 100040, China (e-mail: csewmf@zju.edu.cn).

Digital Object Identifier 10.1109/TNSM.2022.3173398

threats; 2) Passive defense methods can not effectively deal with the endless security threats. To protect the information system, people will add more protective equipment, refine access policies, and load a unified security equipment management system. People will give administrators greater authority to realize the unified allocation and management of policies. These privileged administrators violate the minimum security principles; 3) Traditional defense methods once controlled by the attacker will become a platform for network attacks. For example, attackers can use the firewall to collect intelligence, and tampering with the virus database can cause system paralysis. The above deficiencies of the protection technologies exist in all the IT systems, including the industrial control systems.

Because of this, active defense technologies for IT systems have gradually become a new research direction.

Active defense: it refers to building a flexible defense system in real-time, which can give early warning before the system intrusion, to avoid, transfer and reduce the risks faced by the information system.

Typical technologies include the active defense and intrusion tolerance method proposed by Huang *et al.* and the mobile target defense method proposed by Huang *et al.* [10] and Zhang *et al.* [11]. The basic principle is that they adopt a joint defense mechanism integrating threat identification, reinforcement, attack detection, monitoring and perception, response and recovery, and dealing with threats in time or in advance. The active immune method based on trusted computing proposed by Shen *et al.* [12] and Wang *et al.* [13] mainly deploys trusted computing chips and protocols on operating sites, boundaries, and networks. They adopt active identification and integrity measurement strategies to ensure the system consistently executes a correct operation, and then unknown threats cannot infect the system. The defense method based on mimic calculation proposed by Hu *et al.* [14] mainly uses a redundant mechanism and deploys diverse components, network structures, and system access methods on the target system, making the attacker's prior knowledge invalid.

The active immune method's four core technologies based on trusted computing [12] are the optimized deployment of trusted platform modules (TPM), multi-level access control, dynamic integrity measurement, and trusted management of the control network. TPM is the environment foundation of active defense, which relies on cryptography to ensure it can't be tampered with. Access control and integrity measurement are the basic defense technologies for operating systems and boundaries. Trusted management is the core mechanism of active defense at the network level. This article focuses on the trusted management of the industrial control networks. The trusted management solution based on platform integrity measurement and remote attestation [12] has some problems when it is applied in industrial control networks. 1) The trusted verification of communication objects cannot detect an exception that legal items may perform non-compliance behaviors; 2) it lacks a tolerance mechanism. The rejection of untrustworthy key processes may affect the availability of industrial control systems.

The trust management mechanism draws on human society's trust relationship to characterize computer network nodes' credibility. It can avoid malicious nodes participating in services and ensure that trusted objects execute trusted data streams, thereby improving the network's immunity. The semantic concept of trust can seamlessly transmit in heterogeneous distributed systems and has better flexibility and scalability than traditional access control technologies. The trust mechanism is the core support of the trusted management. There are many works on trust management in other information systems, such as Peer-to-peer (P2P), grid, Wireless Sensor Networks (WSN), Ad hoc, cloud computing, edge computing, and a large number of trust models that accurately describe the network credibility and conform to environmental requirements have been proposed [15]–[24]. However, there is very little work on trust management for industrial control systems. The current researches mainly focus on trust network architecture [25]–[26], the simple application of the trusted computing of ordinary information systems into the industrial control systems [27]–[28], or the limited dimensional trust modeling [29]–[32]. Due to the heterogeneous and strong real-time features of industrial control networks, trust management in the industrial control networks will face a series of problems, including:

1) The identification of malicious behaviors needs to consider the characteristics of cyber-physical integration. Attacks in industrial control systems are often infiltrated from the information side to the physical side or vice versa. It is necessary to identify which abnormal operations may damage physical equipment. The abnormal operations may manifest as unauthorized access to control instructions or non-compliant operations. Malicious behavior identification is lacking in current trust management algorithms;

2) The trust modeling needs to consider the behavioral differences of information operations oriented to the control target, e.g., for crucial instruction operations, the writing should be more sensitive than reading;

3) The availability requirements of industrial control networks are extremely high, and the availability is reflected in the delay and connectivity. Trust modeling should choose lightweight algorithms, where the deployment needs to consider different availability requirements of the different network layers, and the trust decision-making mechanism should ensure that critical services are not blocked.

This paper proposes a trust management method against abnormal behaviors in the industrial control networks under an active defense architecture. The core idea of the method is to identify abnormal behaviors for control instructions in industrial control networks and perform trust evaluation of these behaviors, and effectively deal with malicious behaviors of abnormal nodes based on their trust values, without affecting the availability of the control network. The contribution of this article is as follows:

(1) We review the trust management mechanism in the industrial control networks. We focus on control operations security, and firstly define the basic concepts of trust, security, reliability, and availability of industrial control networks. Then we analyze the application difficulties of trust management in

industrial control networks, including malicious behavior identification, differentiated modeling of critical behaviors, and key operation continuity. And then, we discuss the availability constraints of industrial control networks after applying trust management and give the necessary process and decision-making methods of trust management in industrial control networks;

(2) We analyze the unknown threats taken from the "Stuxnet" and "Ukraine's power outages." The attack's essential operation is the tampering, blocking, and malicious feedback of control instructions on the industrial control networks. We take the instruction's operation behavior as the protected object and extract and model trust information for abnormal behavior. Compared with the existing trust methods, the modeling process considers the difference between malicious operations and adopts the punishment factor to reflect the different importance of the operation.

(3) To ensure the continuity of the industrial network's operations, we establish a trust update and decision-making mechanism under the availability constraints. We reduce the calculation range and reduce the recommendation algorithm's complexity on nodes with high real-time requirements; we use trust compensation and audit response method to ensure network connectivity when the critical nodes' trust value is lower than the threshold.

(4) Taking the distributed control system as an example, we display a deployment method of trust management. According to the different availability of components at each level of the industrial control system, the trust server pushes differentiated trust calculation methods to the trust engine deployed on the target nodes. Trust management mechanism, combined with the Trusted Platform Module (TPM, which comes from the best practice from Ref. [26]–[28]) and audit response method, can realize industrial networks' active immunity. Detailed analysis and simulation also confirmed the effectiveness of the proposed scheme.

The sections of this article are arranged as follows: Section II is the related works, Section III summarizes the trust management mechanism in industrial control networks, and Section IV proposes the trust model of industrial control networks, including abnormal behavior recognition, trust modeling and calculation, and trust updating and decision-making under the availability constraints. Section V presents the deployment of trust management in an industrial control network. Section VI conducts analysis and simulations. The discussion and summary are in Section VII.

## II. Related Works

The current works on the trusted protection of industrial control systems mainly include the research of trust network architecture [25]–[26] or the simple application of the trusted mechanism from common information systems [27]–[28], or the finite dimensions of trust modeling [29]–[32]. Okhravi and Nicol [25] proposed a security framework based on authentication for process control networks according to the concept of trusted networks and discussed the architectural requirements of components, protocols, operating systems, and availability, but this method lacks specific protection strategies. Pinto *et al.* [26] introduced ARM TrustZone technology as a reference security protection method for the device in the Industrial Internet of Things and enhanced the trusted execution environment to meet real-time requirements. However, this method does not consider the security protection of the control network. Harshe *et al.* [27] proposed a trusted autonomous architecture to protect trusted components in programmable logic, and isolate and monitor physical processes from untrusted components. It can detect reconfiguration attacks on the networks and software of the industrial control system. Göttert *et al.* [28] proposed a distributed trusted neighbor discovery protocol (TND) based on trusted hardware and a provable security method. This protocol can be used to monitor, detect and locate attacks, such as software configuration and control sequence changes in industrial control systems. Both Harshe and Göttert's schemes are a direct application of trusted computing into the industrial control systems. The previous Section has analyzed some defects of these application modes.

Fadul *et al.* [29]–[30] applied the trust theory to protect the smart grid's SCADA system. They used the trust mechanism to filter malicious nodes in the smart grid, identified and mitigated smart grid faults, and designed a robust and configurable trust management toolbox. When the trust mechanism is directly applied to the industrial control networks, we need to consider the credibility evaluation of different interactive behaviors and adaptability to the control system. Moreover, Fadul's solution has only focused on the identification and protection of terminal equipment. Zeng and Chow [31] proposed a reputation-based distributed control method to identify the internal misbehavior node in a distributed control system, but it can only recognize misbehaviors for controlled objects - robots. Boudagdigue *et al.* [32] migrated the trust management into a new hybrid architecture constituted of industrial communities. Each community is monitored by a trusted leader who calculates the trust of nodes according to three metrics of cooperation, direct and indirect honesty, and returns the results to the Industrial Internet of things (IIoT) server. The above schemes have not considered the characteristic of availability priority for industrial control systems in trust modeling.

It is necessary to build a universal trust evaluation model to deal with the complex calculus logic associated with dynamic trust evaluation. Some trust evaluation models have been proposed. Representative ones include the trust model based on probability theory [15]–[16], trust model based on entropy theory [17], trust model based on fuzzy theory [18], trust model based on D-S evidence theory [19], the trust model based on semi-ring algebra theory [20], the trust model based on cloud theory [21], etc. These trust models are suitable for some application scenarios, such as P2P [15], Ad hoc [17], [20], WSN [18], [21], cloud computing [22]–[23], Internet of Things [24], [33]–[35], fog computing [36] and no specific scenario [37]. Some other works focused on the application of trust mechanisms in improving the network performance, including trust-based networks with optimized computing, communications, and caching proposed by He *et al.* [38], the trust-based task assignment method with

TABLE I
PERFORMANCE OF TRUST MODELS

| Reference | Applicable scenarios | Characteristic | Trust attribute | Behavioral differences | Availability |
|---|---|---|---|---|---|
| **Ref.[25]** | ICS | Propose trusted security architecture | × | × | √ |
| **Ref.[26]** | Industrial internet | Introduce a trusted execution environment for device | × | × | √ |
| **Ref.[27]** | ICS | Implement trusted components in programmable logic | × | × | × |
| **Ref.[28]** | ICS | Trusted secure protocol | × | × | × |
| **Ref.[29]** | Smart grid | Trust management for SCADA system in smart grid | × | √ | √ |
| **Ref.[31]** | ICS | Consistency detection of robots based on trust | × | √ | × |
| **Ref.[32]** | IIoT | Hierarchical centralized trust evaluation | √ | × | √ |
| **Ref.[15]** | P2P | Bayesian network-based trust modeling | √ | × | √ |
| **Ref.[17]** | Ad hoc | Entropy theory-based trust modeling | √ | × | × |
| **Ref.[18]** | WSN | Fuzzy fully distributed trust computing and prediction | √ | × | √ |
| **Ref.[22]** | Cloud computing | Trusted service selection for hypergraph optimization | √ | × | × |
| **Ref.[24]** | IoTs | Context lightweight recommended trust model | × | √ | √ |
| **Ref.[33]** | IoTs | Anti-attack trust management method | √ | × | × |
| **Ref.[37]** | Undefined | Triple attribute-based Trust modeling | √ | × | × |
| **Ref.[41]** | 5G edge network | Active and verifiable trust mechanism | √ | × | √ |
| **Ref.[44]** | Edge computing | Active and verifiable trust evaluation method | √ | × | √ |

multi-objective optimization proposed by Wang *et al.* [39], and trust-based service management method proposed by Chen *et al.* [40]. Each trust model has differences in behavior identification, trust definition, and conformity with human society. Existing trust methods have not combined with the characteristics of the industrial control networks. The behavior differences are little considered. The problem of unintentional shielding of key operations needs further consideration.

Huang *et al.* [41]–[42] proposed an active and verifiable trust acquisition mechanism in the intelligent edge network. The data center dispatches unmanned aerial vehicles (UAV) to collect the transaction information from the IoT devices. The data offloaded by the official UAV is treated as the baseline data and formed into a trust verification chain to verify the data provided by mobile vehicles (MVs). In [42], Huang *et al.* further proposed an effectiveness-based incentive mechanism and a secondary path planning strategy to obtain more baseline data and improve the efficiency of trust evaluation. A similar method is a trust-based active detection (TBAD) scheme proposed by Liu *et al.* [43], the UAV evaluated the trust of sensor nodes based on the reliability of data packets, where the trust value of the node is stored in the header of data packets. The sensor nodes with higher trust are selected to form movement trajectories. Mo *et al.* [44] proposed another active and verifiable trust evaluation method for edge networks. They required the receiving node to return the encoded information of the transmitted packet for verification. The length of the encoding is the same as the information length of a packet. By this method, the nodes on the path can establish a trust relationship, and malicious nodes (e.g., dropping packages) can be found. A similar mechanism includes a two-hop feedback method proposed by Wang *et al.* [45]. These methods are reflected in the collection of trust information, and there is no active architecture similar to trusted computing. Its network structure and node mobility are not suitable for the industrial control system. Some typical trust mechanisms are shown in Table I.

In Table I, "√" indicates that it has a related property, and "×" means it does not have. The three columns on the right are the trust management requirements in the industrial control system. The trust model should accord with trust's attributes while considering the differences in operation behavior related to field devices and the industrial control system's availability.

The existing trust management methods show excellent performance in some aspects, which can be used as a reference. The investigated trust models have differences in trust attributes. For example, for the entropy-based Ad hoc trust model, the sensitivity and transfer ability are good, but the scalability is insufficient. The trust mechanism in Ref. [25] involves availability, but it is only an architecture without specific implementation, and the involved behavior difference in Ref. [29] only considers terminal equipment. The availability in [41], [44] mainly involves the scalability of the trust evaluation method, not the availability of the edge network. Therefore, the trust model conforming to the characteristics of industrial control systems needs further research.

Based on the active defense idea and trusted computing architecture [12], this paper establishes a trust management model on a typical industrial control network in the Purdue model [1], [46]. We identify abnormal behaviors under the influence of unknown threats for the industrial control networks, then we evaluate the nodes' credibility on the operation behaviors based on the trust model proposed in Section IV. We make decisions based on the trust evaluation results, shield or restrict malicious nodes from participating in interactions, and continuously monitor trusted nodes' behavior and their residual energy to ensure the industrial control system's availability and trustworthiness.

## III. THE TRUST MANAGEMENT FOR ABNORMAL BEHAVIOR IN THE INDUSTRIAL CONTROL NETWORKS

### A. Abnormal Behavior in Industrial Control Networks

According to Cheminod *et al.* [1], a typical industrial control system is shown in Fig. 1.

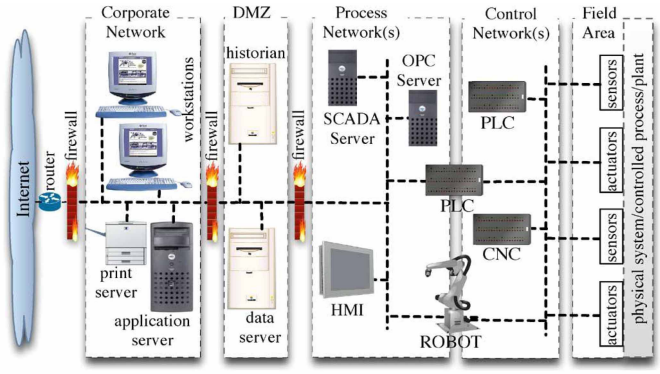



Fig. 1. A typical industrial control system architecture.

The industrial control systems are constructed by a series of heterogeneous devices, including Programmable Logic Controller (PLC), Remote Terminal Unit (RTU), OPC server, Human Machine Interface (HMI), SCADA server, data server, engineer workstation, application server, gateway, firewall, etc. This article focuses on the trusted management of the network from the field level to the Demilitarized Zone (DMZ), which is referred to as "industrial control networks".

*PLC:* It is the most important component to realizing the control function. It accepts the management and instruction download from the monitoring server and controls the field-level equipment according to the programmed process. At the same time, the PLC collects the data from the field sensors and uploads the data to the monitoring server for display and possible policy adjustment. The biggest threat of PLC is the non-compliant operation or mis-operation.

*RTU:* It is a remote terminal unit, which is mainly used for remote measurement and control. It can issue simple instructions to sensors and actuators within the monitoring range, and upload the data collected by sensors to the SCADA server after simple processing. The biggest threat of RTU is unauthorized capture and blocking.

*OPC server:* OPC is one of the data connection standards used to communicate between PLCs, monitoring servers, workstations, applications systems, etc. The OPC server provides interoperable interfaces between different industrial protocols, devices, and systems. The main threat lies in unauthorized access.

*SCADA server:* It realizes the functions of data acquisition and monitoring control. It connects PLCs and RTUs in a wide area, realizes centralized management of large-scale control equipment, and can also issue control instructions to the controller. Unauthorized operations and non-compliant operations will affect its security.

*HMI:* It is a medium for interaction and information exchange between the control system and users. It realizes the conversion between the internal form of information and the form acceptable to humans. The HMI is generally deployed on the control server. Non-compliant operations and information blocking will cause false displays.

As for data servers, engineer workstations, application servers, gateway, firewalls, the form of these devices is similar to that of ordinary information systems, but in the industrial control scenarios, they support industrial-specific protocols, industrial embedded applications, real-time data reading and writing of industrial data.

When the industrial control system runs, each device will implement a group of normal operations on the information flow according to the procedures programmed in advance. The most important operation is executing and transmitting the controlling instructions. The workstation initiates instructions and transmits them through the application server, gateway, controlling server, controller, and field devices. These devices will perform operations on the instructions, such as authentication, transmission, calculation, storage, and if necessary call the data server, to ensure its correct and continuous implementation; the compliant modification of the control instructions according to the feedback information from the field device also undergoes similar operations from controller, host computer, server, etc.

However, industrial control network is susceptible to external attacks and internal malicious behaviors, resulting in abnormal behaviors.

*Unknown threats* refer to an attack that cannot be observed or predicted in terms of threat source, threat type, threat time, and available vulnerabilities.

The attack surface of the industrial control system is large, and different nodes at different levels may suffer external or internal attacks. The attack method can be traditional software failure, buffer overflow, or information physical penetration, such as error data injection. The time point of launching the attack is uncertain, some attacks may hide for a period. There are undetectable vulnerabilities in industrial control equipment, protocols, and businesses, which may be continuously exploited.

The common implementation method is the advanced persistent threat (APT) [47]. This attack flexibly combines various new attack methods to penetrate the target for a long time and implement the attack at a specific moment. The typical attack process for industrial control systems includes five stages: information collection, breakthrough the defense line in the border, establishing strongholds and penetrating horizontally, attacking the industrial server, disrupting the field control logic, or interrupting the service, and then completing the attack. The "Stuxnet" virus and "Ukrainian blackouts" adopt a similar method. The "Stuxnet" virus invades the nuclear power control system through social engineering methods. It uses device vulnerability to gain the control system's operating authority and realizes the centrifuge's stall control. In the "Ukrainian blackouts", hackers intruded into the host computer by operating the malicious software (BlackEnergy) downloaded to the employee's workstation, and directly issue the power outage command through the SCADA system. At the same time, they also interfered with the power company's telephone communication, resulting in that the residents affected by the shutdown could not contact the power company.

These attacks exploit the vulnerability of industrial control systems, such as equipment vulnerability, protocol vulnerability, and business vulnerability. This paper focuses on business vulnerability.

Business vulnerability refers to the vulnerability that the normal control logic and key data may be disturbed and tampered with. For example, the attacker may interrupt or temporarily block the control process, during which the field devices are still running, but they are not necessarily in the expected state; the attacker modifies the control instructions and operates the physical equipment in a non-compliant manner; the attacker tampers with the view and guides the operator to issue wrong control instructions; The attacker uses denial of view to prevent the operator from monitoring the site, resulting in abnormal status that cannot be noticed; The attacker steals the operation information in the production environment.

Unknown sources may attack the industrial control process carrying the business and may damage the availability, integrity, and confidentiality of the control process. Unknown threats can lead to three types of typical abnormal behaviors related to a business vulnerability in the industrial control networks, including:

1) *Unauthorized access to control instruction:* It is the most typical abnormal behavior. For example, the "Stuxnet" virus modified the control instructions and caused the centrifuge to stall. The ultimate goal of an attack is to modify control instructions. The attacker injects control commands by sneaking into the control server and control links or induces control objects, e.g., sensors, to deliver wrong control commands. Devices on the link of transferring instruction may undergo unauthorized access, tampering, or disclosure. These devices may be malicious or non-malicious. Some devices may participate in transmitting malicious instructions without their knowledge.

2) *Non-compliant operation of control instructions:* It is induced by internal abnormal or malicious behavior. It includes authorized users' non-compliant interaction, modification, interception, deception, discarding, etc., and the mis-operation of legitimate operators. Some devices may drop packets, selectively forward packets, and replay packets. The Canadian sewage discharge incident in 2007 was caused by deliberate operation by insiders.

3) *Interfering with normal operations:* Some devices may block information transmission or send a large number of packages to the same address. These behaviors may disrupt normal operations and make it impossible to provide services reliably. It may be malicious or caused by insufficient energy of the remote node. For example, the "Ukrainian blackouts" is an attack completed by disturbing regular judgment upwards and modifying control instructions downwards.

From the attacking process of "Stuxnet" and "Ukraine Power Outage," it can be seen that the goal of an attack is to disrupt the industrial site environment. The critical step is to block and tamper with control instructions. This procedure is accompanied by some abnormal operations for the control instructions. Therefore, identifying, shielding, and blocking these abnormal behaviors in the industrial network in time and maintaining the industrial network's regular operation is the primary idea of active defense such as trusted computing. Trust management is one of the important methods to avoid the occurrence or spread of abnormal behavior. It cooperates with TPM (Trusted Platform Module) and audit mechanisms to make the industrial control system immune to attacks.

### B. Trust Management Mechanism of Industrial Networks

Under the active immune architecture [12], we give the trust definition for anomaly identification and active defense of business operations in the industrial control network.

*1) The Definition of Trust and Reliability:*

*Definition 1 (Trust):* Trust is the subjective expectation and dynamic cognitive process of the object's ability to provide the services to the subject. Its connotation includes the service provider's reliable belief and risk assessment to complete the task in a specific context.

*Definition 2 (Reliability):* The ability of the industrial control system to provide services continuously, R(t) represents the probability that the system continues to provide services until $t$. Let the random variable X be the time of unexpected failure of the system, and the reliability is expressed as $R(t) = Pr(X > t)$.

*Definition 3 (Security):* Security means it can avoid unauthorized access and unauthorized or accidental changes, destruction, or loss of system resources. And it can prevent illegal or harmful intrusion into industrial control systems or interfere with its correct and planned operation.

*Definition 4 (Availability):* The control process and resources are available. The system is at the probability that the specified or restored function can be performed under the specified conditions and within the specified time or time interval.

If the industrial control system performs as expected, the industrial control system is trustworthy, which means that it can continue to provide services and be immune to attacks while ensuring that legitimate users' legitimate behavior is not blocked. That is, trust is related to reliability, security, and availability. Trust management is a concrete realization method to ensure a network's trustworthiness.

*2) Trust Management and Its Application in Industrial Control Networks:* Trust management is a method that completes a decision based on the service object's trust status. Through the trust mechanism, trusted nodes and untrusted nodes can be identified. Untrusted nodes come from two aspects: external attacks and internal malicious behaviors. The trust management is to establish a consensus cooperation mechanism for nodes in the industrial control networks and to be able to select a trusted trust chain to perform tasks; at the same time, during the interaction, a fine-grained access control mechanism based on the trust degrees can be established.

The accuracy of trust management depends on trust measurement, including behavior identification, definition, and trust modeling. Different mathematical models and methods have been proposed, e.g., trust modeling based on probability theory, fuzzy theory, DS evidence theory, semi-ring algebra theory, etc. The identification and definition of behavior are described in their theoretical scopes, e.g., the probability-based trust method is mainly the probability calculation and prediction based on the number of successful and failed transactions. The conformity of trust attributes to human society

has become an important direction for the trust models, such as precise reasoning of cognitive ambiguity, reward and punishment mechanism of trust, time decay of trust, etc. The structure and requirements of different networks are inconsistent, and the applicability of the trust model is also an important research direction. For example, trust management methods are required to have good scalability in P2P and grid environments. WSN, the Ad hoc environment requires trust management have good sensitivity (timeliness of trust change) and lightweight deployment.

Due to heterogeneous and strong real-time characteristics, the trust management for the industrial control networks has some differences from other scenarios:

(1) The identification of malicious behaviors needs to consider the characteristics of the cyber-physical integration scenario. It is necessary to identify which kind of abnormal operations may damage physical equipment. The irregular operations may manifest as unauthorized access to control instructions, non-compliant operations of the controlled device, etc. Existing trust management algorithms lack malicious behavior identification;

(2) The modification and illegal control of the controlled object is the goal of the attack on the industrial control system. It should reflect the behavioral differences of operations (such as essential instruction operations) related to the control target. For example, the instructions writing should be more sensitive than the reading, which can be reflected in the trust modeling;

(3) The trust management algorithm should not affect or minimize the impact on the industrial control system's regular operation. The availability is reflected in the delay and connectivity. The trust management method should choose lightweight algorithms, its deployment should consider the availability of different control layers, and the trust decision-making mechanism should ensure that critical services are not blocked.

*3) Availability Constraint of Trust Management in Industrial Control Networks:* The availability constraints in the application and deployment of trust management in the industrial control systems mainly include the following aspects:

*Latency:* Industrial control networks require high real-time performance. It requires the services to be completed with extremely low latency. According to engineering experience, the business in common industrial control scenarios is at the millisecond level. The deployed trust management algorithm should be within the delay range. The computing and storage resources occupied by the collection, calculation, and decision-making of trust will affect the delay. Besides taking advantage of the industrial control network's heterogeneous characteristics, it is also a feasible method to deploy trust mechanisms with different complexity on components with different real-time requirements.

*Connectivity:* Industrial control systems must ensure the continuity of key operations. The fundamental decision-making of trust management is to select a trusted link to provide services and shield untrusted nodes. If a node is undertaking control tasks, such as sending, receiving, and transmitting key control instructions. It should not be immediately blocked by the trust mechanism, even if its current trust value is lower than the threshold. We should take temporary measures (such as adjusting the trust value to a threshold and restricting access to the next nodes) to ensure the smooth transmission of crucial commands.

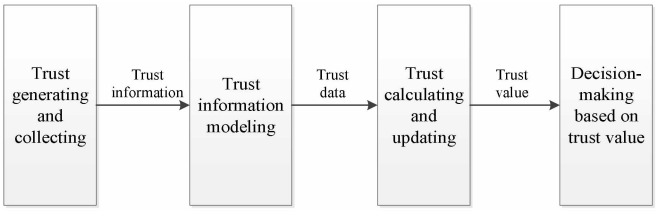



Fig. 2. Process of trust management.

*Recovery time:* Reducing recovery time can improve availability. As an additional measure to the trust framework, recovery measures are initiated when the critical node is not trusted, including starting the standby node and submitting it to the quick audit module.

*4) Trust Management Process in Industrial Control Networks:* Without losing generality, trust management includes four parts: the generation and collection of trust information, trust information modeling, the calculation and update of trust, and the decision of trust, as shown in Fig. 2.

Trust is the feedback of the node interaction process and results. After two nodes interact, the object judges the interaction process and results subjectively and determines the service satisfaction of the subject from his perspective. Sometimes it is necessary to use fuzzy theory to calculate multi-dimensional vague satisfaction quantitatively. The satisfaction series within a time window is the original data set for trust modeling. The trust engine residing in node $A$ collects the satisfaction sequence with the target node $B$, including the historical sequence of direct interactions between node $A$ and node $B$, and the historical sequence of interactions between other adjacent nodes in the target network and node $B$. Then the trust engine uses mathematical methods to model trust, and then calculates trust values, including direct trust values, indirect trust values, and other attribute values (such as rewards and punishments). And finally, the trust engine obtains the overall trust value of the target node. The trust server and each engine repeat all the processes to obtain the trust degree of all nodes in the target network, and then make decisions based on trust.

The trust decision of the industrial control system includes the following three aspects:

*1) Fine-Grained Access Control:* Different trust degrees of nodes determine different rights to accessing resources. Gradually increasing the importance of operations, such as transfer, read, and write the control instruction, correspond to gradually increasing access rights. Nodes with higher trust values can perform more important operations.

*2) Path Optimization of Control Flow:* Based on the target network nodes' trust values, the service requester can select a credible path to transmit data according to the optimized constraint of link length, energy requirements, and delay. The trust

decision for path optimization in general information systems is also applicable in industrial networks.

*3) Linkage With Detection and Auditing:* The trust manager needs to submit the untrusted nodes or behaviors to detection modules, to audit whether they are intruded, in real-time or semi-real-time. Auditing and recovery are essential for scenarios with high availability requirements.

The trust mechanism monitors network nodes' behavior in real-time, allows normal operations, authenticates, blocks or shields abnormal operations, and makes malicious actions invalid. The next section will elaborate on the trust model of industrial networks based on the trust management process.

## IV. The Trust Model of Industrial Control Networks

### A. Abnormal Behavior Identification of Industrial Networks

We deploy a trust management server in the industrial control networks and deploy a trust computing engine on the calculation node to collect information among nodes.

The collected information includes historical information (history behavior sequence recorded) and current recorded information. The history data is used to calculate the behavior node's current trust value; the current information is used to update the trust. The server defines a time window $T$ of the control flow and collects the original record of the interaction behavior among nodes in the time window.

Normal operations are recorded in order by time window. The key is the record of abnormal operations. Abnormal behavior mainly includes three aspects.

*1) Unauthorized Operation of Control Instruction:* The devices on the instruction transmission link may perform unauthorized access, tampering, and leakage. It includes two situations: the control instructions flow through unauthorized nodes or processes; instructions are operated by unauthorized users or processes (read, write, communicate, process, etc.).

*Abnormal judgment:* The trust computing engine resident in the node records these abnormal behaviors, and it tracks the previous or next hop of the neighbor node with the abnormal behavior on the instruction transmission link, and the record is fed back to the trust management server in a certain period.

*2) Non-Compliant Operation by Legal Operators:* It includes the authorized object performing non-compliant interaction, modification, interception, spoofing, discarding control instructions; the mis-operation by legal operators; disrupting the trust mechanism, such as packet loss, selective forwarding, replay, deception, malicious feedback, etc.

*Abnormal determination:* The trust engine detects non-compliant interaction, modification, deception, discarding, and mis-operation by authorized users; it detects that the resident node or its neighbor nodes (including multi-hop neighbor nodes) have the behavior of blocking information transmission, packet loss, selectively forwarding and replaying packets, all of these are abnormal. The records are fed back to the trust management server in a period.

*3) Interference With the Regular Operation of the Control Instruction:* Someone may misconfigure the protective equipment, making it impossible to pass the key commands; someone may send a large number of packages to the same address, disrupting normal business operations, and making it impossible to provide services reliably.

*Abnormal judgment:* The trust computing engine monitors that the control instructions are intercepted abnormally; it can observe that the data packet's destination address continues to point to a particular resource-sensitive node; the above behaviors are abnormal.

TABLE II
Abnormal Behaviors

| Abnormal type | Abnormal content | affected objects |
|---|---|---|
| **Unauthorized operation of control instruction** | Control instructions flow in unauthorized nodes or processes | Nodes on the link |
| | Control instructions manipulation in unauthorized nodes or processes | Self or neighbor node |
| **Non-compliant operation of control instruction** | Non-compliance manipulation of control instructions in authorized nodes | Self or neighbor node |
| | Disrupting the trust mechanism or malicious data packet manipulation | Self or neighbor nodes |
| | Mis-operation of control instructions | On-site control node |
| **Interfering with normal operations** | Control instructions are intercepted or blocked | Neighbor node |
| | Data packets attacking resource-sensitive nodes continuously | Nodes on the link |

TABLE III
Trust Information

| Node identity | Time window | Initial trust | Operational behavior sequence | Operation access threshold |
|---|---|---|---|---|
| $Id$ | $T$ | $T_0$ | $R_1, R_2, \ldots, R_n$ | $Th_1, Th_2, \ldots, Th_n$ |

The possible abnormal behaviors are shown in Table II. The third column is the affected object, which means abnormal behaviors may occur on these objects, and the trust computing engine resident on these objects can recognize abnormally.

### B. Trust Modeling of Industrial Control Networks

In a time window $T$, we observe the interaction process of nodes for operating control instructions in the industrial network and establish behaviors trust for the nodes. Trust is a subjective judgment of whether a node's behavior is credible. How assess whether a node is trustworthy from its neighbors is the primary procedure of trust modeling.

If the node always operates as expected during the time $T$, then the node is trustworthy; if the node always behaves abnormally, the node is not trustworthy; the actual situation is that there is both normal operation and abnormal behavior, and the state of the node is uncertain, it need to quantify the trust degree. First, we give the trust information as shown in Table III.

*Identity:* It indicates the entity of trust. This article mainly refers to the subject and the object.

*Time window:* It refers to the period in which trust information is collected and calculated. The time window is sliding. Each time it slides for periods $(\Delta T)$, the records that slide out of the window will not be involved in calculating the trust value at the current time as trust has the attribute of time decay.

*Initial trust value:* It is the trust value assigned when an entity joins the control network. Too high an initial value may easily lead to malicious attacks. Too low a value may cause some nodes to have no chance of a transaction. Set the trust value range to [0, 1], and generally the initial value $T_0 = 0.5$.

*Behavior sequence:* The operation behavior sequence of an entity, denoted as $R_1, R_2, \ldots, R_n$, it corresponds to a set of trust level sequences $T_1, T_2, \ldots, T_n$, after the trust calculating.

*Behavior threshold:* It is the access control thresholds of operation, expressed by $Th_1, Th_2, \ldots, Th_n$, whether a set of operations are allowed are judged by these thresholds, for example, if $Th_i = 0.8$, a request with a trust value between 0-0.79 is blocked.

*Give an example of trust modeling:* Node *A* performs a set of operations in a period *T*. Each group of operations (*e.g.,* calculation data, communication, processing instruction) corresponds to an interaction between two nodes (node *A* and his neighbor node *B*). Then node *B* evaluates the operational effectiveness for each interaction. It can determine whether the interaction was successful or not based on abnormal recognition. At the same time, the historical interaction records are examined. The trust engine in node *B* chooses a method, calculates, and updates the trust value. Then it compares with the access threshold to determine whether the node is trusted currently. A group of interactions will be calculated sequentially in period *T* in the same manner. Then the trust value sequence is obtained.

We deploy a trust calculation engine on each node to calculate and store trust values. The engine collects historical interaction sequences of entities within a period *T*, calculates the current trust value after a fixed period or when abnormal behavior occurs, and updates the trust value according to whether there is abnormal behavior.

Trust calculation mainly reflects the functional attributes of trust and its compliance with the trust of human society, such as subjectivity, fuzziness, time attenuation, reward and punishment mechanism, robustness, sensitivity, transitivity, scalability, and so on. Without losing generality, we calculate the global trust value by the weighted average of direct trust and indirect trust [38], [40]. We also consider the reward and punishment mechanism, and the number of interactions. We calculate the trust value of node *i* to node *j* at the current time *t*, as shown in (1).

$$T_{ij} = \begin{cases} \frac{\alpha(t)}{\alpha(t)+\beta(t)} \cdot DT_{ij} + \frac{\beta(t)}{\alpha(t)+\beta(t)} \cdot RT_{ij} + RW(t), & N_d(t) < Th_n \\ DT_{ij} + RW(t), & N_d(t) \geq Th_n \end{cases} \tag{1}$$

where, $\alpha(t)$ and $\beta(t)$ are the adjusted weights of the direct trust value and the recommended trust value, respectively. The weights of the direct trust value and the recommended trust value reflect the adopted proportion to the direct trust and the recommended trust by the current node in the time point *t*. For example, when there are fewer connections between nodes or in idle time, the trust engine will collect more indirect trust information, and $\beta(t)$ will be increased. In the stage of frequent interaction, the trust engine will accept more direct interactive information, and $\alpha(t)$ will be increased. $\alpha(t)$ and $\beta(t)$ can be adjusted dynamically. *RW*(*t*) is the reward trust value. The reward value is determined by the node's continued popularity, that is, if a node continues to provide good services within a period, its trust value should be higher than that of non-continuous providing good services, and the increase of trust value should be small enough to not affect the punished trust value (i.e., the trust value is reduced by providing malicious services) to avoid on-off attacks. Therefore, within an observation time *T*, the reward value is $RW(t) = 0.1\ N_{suc}(t)/N(t)$, $N_{suc}(t)$ is the number of consecutive successful interactions until the current time *t*, *N*(*t*) is the total number of interactions until *t*, that is, the reward value is controlled below 0.1, to prevent the counteraction of punishment. When the direct interactions $N_d(t)$ reach some times $Th_n$ (time threshold) during the investigated time, it can be considered that the two parties completely establish a trust relationship. At this time, only direct trust is considered, which is consistent with human cognition habits.

The interaction between the entity and its neighbor entity belongs to direct interaction. According to the interaction result, the object evaluates the subject. If it is a successful interaction, the number of successful interactions (normal operation behaviors) increases by 1. Otherwise, the number of failed interactions increases by 1. Abnormal operation behaviors mainly cause failed interactions, and a small part of failed interactions is caused by disturbances or accidental failures.

Subjective logic is a method for probabilistic information fusion, it uses the opinion to denote a subjective belief and can model positive, negative statements and uncertainty. The subjective logic method has been widely adopted in trust management [48], [49]. Since the subjective logic method is lightweight and can reflect the subjective and statistical characteristics of trust, we use it in direct trust modeling for industrial network nodes. The direct trust value of node *i* (in domain $Q_1$) to node *j* is calculated by the subjective logic theory, that is:

$$DT_{ij} = b + \frac{1}{2}u = \frac{s}{s+f+1} + \frac{1}{2}\left(\frac{1}{s+f+1}\right), \tag{2}$$

where *s* and *f* represent the number of successful interactions and failed interactions (including abnormal behaviors) up to the current time, respectively.

We obtain the indirect trust value $RT_{ij}$ by the weighted average of the historical reputation sequence of node *j* interacting with the node-set in the same domain (set as $Q_2$) in the same period *T*.

$$RT_{ij} = \frac{\sum t_{ik} t_{kj}}{\sum t_{ik}}, k \in Q_2, \tag{3}$$

where $t_{ik}$ is the trust value between nodes *i* and *k*, $t_{kj}$ is the trust value between nodes *k* and *j*, the recommended value is essentially the direct trust value between the third-party node and node *j*, and the calculation method is equivalent to the

calculation of the direct trust value. When there is no direct interaction between node *k* and node *i*, we perform the iterative calculation as described above. In general, for nodes in which the hops are too far away from *i* or *j*, the interaction opportunities and interaction types (generally only the potential demand for transmission) are limited in the industrial control networks, and their recommendations are not significant reference to the target node. Moreover, the calculation amount will be increasing. Therefore, we consider the recommended nodes within 2 hops generally. We only calculate the direct trust value for field devices or nodes with high real-time requirements.

The trust value calculated according to formula (1) is the current trust value between node *i* and node *j*, and is temporarily stored in the current computing node (node *j*) together with the historical trust record, recorded as $\{TV_1, TV_2, \ldots, TV_n\}$.

Similarly, we calculate other nodes' trust values in the same manner to obtain the trust values of all nodes in the target network.

### C. Trust Updating of Industrial Control Networks

In trust updating, we will consider the industrial control networks' characteristics and the differences in abnormal behaviors.

If there is no abnormal alarm, the trust value is updated at a fixed period *T*. If there is an abnormality, the trust engine extracts the log information, including the abnormal time, abnormal node location (self, neighbor, or node in the link), abnormal type (unauthorized instruction write/read, non-compliant write/read, interfere with regular operation), operation object (key instructions, stream data, auxiliary software, etc.), to determine the abnormal level. According to the requirement that availability $>$ integrity $>$ confidentiality and the different priority of the critical abnormal actions (e.g., unauthorized operations include writing, processing, tampering, communication, the writing is more critical than communication), we can determine the degree of threats of the abnormal behavior described in Table II (unauthorized instruction operation, non-compliant operation, interfere with regular operation). Besides, the node's location or the distance from the core controller is also an essential parameter of the threat degree. Therefore, the threat degree of the abnormal behaviors is related to the abnormal type, operation object, node's location, etc.

For example, the PLC controller belongs to the core node. The 1-hop distance is defined as the node with the verification function that is closest to the core node. If each node has an abnormality verification module, then the 1-hop distance node is the geographical neighbor node. We set the threat level of the abnormality to 1-3 (3-serious threat, 2-relatively high threat, 1-general threat). A serious threat is defined as "an abnormal operation that can cause direct, immediate, irrevocable and serious damage to the control logic." The relatively high threat is defined as "abnormal operation can cause indirect, non-immediate and controllable damage to the control logic." The general threat is defined as "an abnormal operation that may cause general damage to the control object."

TABLE IV
RELATIONSHIP OF IRREGULARITIES AND THREAT LEVELS

| Abnormality | Core node | 1 hop distance | 2 hop distance | 3 hop distance |
|---|---|---|---|---|
| **Noncompliant writing/ mis-operations** | 3 | 3 | 2 | 1 |
| **Unauthorized writing** | 3 | 3 | 2 | 1 |
| **Noncompliant reading** | 2 | 2 | 2 | 1 |
| **Unauthorized reading** | 2 | 2 | 1 | 1 |
| **Unauthorized transmission** | 2 | 1 | 1 | 1 |

Taking the unauthorized operation and non-compliant operation of the control command as an example, the relationship between threat level and abnormality is shown in Table IV.

Note that the mentioned typical abnormal operations are considered in the penalty factor. Other abnormal operations in Table II are suitable for different measures. For example, the blocking and interception of control instructions are the most serious case. The trust engineer should directly isolate and repair non-important nodes and repair critical nodes immediately under the premise of ensuring connectivity. Large data packets attacking resource-sensitive nodes and disrupting the trust mechanism is a routine failed transaction. The engineer will update exception times and substitute them into formula (1) to calculate the trust value without further punishment.

Assuming that a node has an abnormal behavior, it is necessary to comprehensively consider the above situations in Table II and speculate on its threat level. Because of the difference in the subjective judgment under the multi-factor fusion, we use fuzzy inference methods to calculate the threat level. Let the discourse be $U = \{$abnormal behavior set$\}$, and set the target domain $V = \{$0-low threat, 1-general threat, 2-relatively high threat, 3-serious threat$\}$. The membership function is a trapezoidal piecewise function. Given a weight vector $W = \{w_1, w_2, \ldots, w_n\}^T$ for $U$, according to the fuzzy set theory, we can obtain the final threat vector, $V_T = \{v_1, v_2, \ldots, v_m\}$. The threat value of a single abnormality is defined as $S_T = ceil(\max(V_T))$, where max $(V_T)$ is the maximum value of $V_T$, it represents the maximum membership degree of abnormal behavior relative to the threat level, and *ceil*() represents rounding up.

The trust value is updated according to the abnormal behavior after the interaction.

The trust engine modifies its current trust value $TV_k$ of the node, $\Delta T_k$ is defined as the difference between $TV_k$ and $TV_{k-1}$. If $\Delta T_k < 0$, The trust engine modifies the trust value $TV_{k+1} = TV_k + \eta \cdot \Delta T_k$. Otherwise $TV_{k+1} = TV_k + 1/\eta \cdot \Delta T_k$. Where $\eta$ is the penalty factor and $\eta > 1$, it is determined by the abnormal level, and is set to $\eta = S_T$. $TV_{k+1}$ is the updated trust value, which will override $TV_k$. This setting is in line with human communication habits. That is, the increase in trust is slow, the decline in trust is rapid. For malicious nodes, more honest transactions are needed to restore the original trust level.

If the node has a high threat level (i.e., a key node is compromised, the field controller command has tampered, the

blocking and interception of control instructions, *etc.*), the trust computing engine directly sets the trust value of the non-important node below to the untrusted threshold (the minimum resource access threshold). Meanwhile, it notifies the network manager to start the system repair measures.

The trust management mechanism focuses on neighbors' behavior within two hops distance in the industrial control networks and calculates and updates the neighboring circle's trust value related to abnormal operations.

### *D. Trusted Decision of Industrial Control Networks*

The deployed trust management server and trust computing engine mark the nodes' operations behavior and adjust the trust value under normal and abnormal behavior. The decision is made based on the trust values set of neighbor nodes saved by the current nodes.

*1) Path Optimization:* According to the basic principles of network services, when a service request is initiated, ideally, selecting a set of nodes with high trust degrees to provide services will have higher quality or perceived credibility. In an actual situation, we need to consider constraints such as delay, resource allocation, and trust status and choose the relatively optimal path to provide services among multiple trust chains. Moreover, in the industrial network environment, the number of nodes between the behavior initiator and the final receiver is limited, and it is impossible to choose the route freely. For example, in the controlling process, instructions will flow through the engineer's operating station, gateway, HMI, PLC, and actuator. According to the path node's trust value, the delay requirement of the entire operation process, and the constraint of node resource, the shortest trusted path is selected to complete the service.

*2) Access Control:* An essential decision-making method is access control based on the trust value. The core is to determine the trust threshold. In the industrial network environment, the importance of interaction behavior is different. For example, the read and write are more critical than the non-associated communication, and the resource sets accessed are also different. The threshold should be different for different interaction behavior.

To perform fine-grained access control, we define a set of decision functions: suppose that the trust degree is divided into $l$ levels. It corresponds to $l$ disjoint intervals. The threshold set of trust levels is $\{Th_1, Th_2, \ldots, Th_l\}$, corresponding to $l+1$ levels of access rights $S = \{s_0, s_1, \ldots, s_l\}$. Then the decision function between the current trust value $T_R$ of the access requester (that is, the $TV_{k+1}$ of an access requester) and $S$ defines as follows.

$$S(T_R) = \begin{cases} s_l, & T_R > Th_l \\ s_{l-1}, & Th_{l-1} \le T_R < Th_l \\ \vdots & \vdots \\ s_1, & Th_1 < T_R \le Th_2 \\ s_0, & T_R \le Th_1, \end{cases} \tag{4}$$

when a node requests an operation, the trust engine first determines whether the operation is allowed based on its trust value and the threshold. For example, for unimportant communication flow, we can set $l = 1$, $Th_l = 0.5$, if the trust value of the access request node is greater than 0.5, normal access can be obtained, and access is denied when the trust value is less than 0.5. For important operation flow, we can set $l = 3$, $Th_1 = 0.5$, $Th_2 = 0.6$, $Th_3 = 0.8$. If the access rights $S = \{$access is not allowed, communication is allowed, information flow and communication is allowed, all the operation and information flow and communication is allowed$\}$, the corresponding $l = \{[0\ 0.5], [0.51\ 0.6], [0.61\ 0.8], [0.81\ 1]\}$, if the node's trust value $T_R = 0.81$, then the node can access the operation object and operate normally. The trust engine can flexibly set the trust threshold according to the business flow's importance to avoid the system's access risk. The larger the value $l$, the more the trust levels, and the more accurate the access control.

### *E. Trust Decision Adjustment for Availability Constraints*

Network attacks and random failures will lead to unauthorized or non-conforming behaviors of nodes. Trust mechanisms can alleviate or avoid these exceptions. However, whether the deployment of trust mechanisms will cause new problems, such as affecting system availability? Firstly, we give the definition of availability, analyze its influencing factors, and then judge and adjust the trust mechanism. The availability is related to reliability and repair time [50], and it is formulated as:

$$A = \lim_{t\to\infty} E[Y(t)] = \frac{MTTF}{MTTF + MTTR}, \tag{5}$$

$$MTTF = E[X] = \int_0^\infty R(t)\,dt, \tag{6}$$

where $Y(t)$ is the availability probability of the system at time $t$. *MTTR* is the average time to repair. It includes any time when a fault is detected, repaired, and the system is put back into operation. *MTTF* is the meantime before failure, which represents the expected time before the first failure. *MTTF* is defined in (6), and $R(t) = Pr(X > t)$, the random variable $X$ is the time of system random failure. Therefore, reliable running time and repair time are important factors affecting availability.

The impact of the trust decision on availability includes some factors: the energy consumption in a single point, to ensure the reliable operation, the energy consumption of the trust calculation of a single node shall not affect the energy required for the normal control operation in the node; the tolerable delay, the delay of control operation caused by trust mechanism should be within the acceptable range; the decision impact on connectivity, the trust mechanism should not directly block critical control flow; the disposal and recovery time, the processing and recovery time of critical fault nodes should be as short as possible.

*1) The Energy Consumption EC:* Some industrial network nodes have limited resources and have quick response requirements. The sensor with the old version generally has a remaining storage space that is no more than 1M, available bandwidth is no more than 512Kbps.

Let a node-set of the target industrial control networks be $N_R, N_R = \{N_{R1}, N_{R2}, \ldots, N_{Rn}\}$, $n$ is the number of

nodes, each node has a set of available resource $NR_p = \{p_1, p_2, \ldots, p_m\}$, $m$ is the number of resources. The three typical resources are computing power $(p_c)$, storage capacity $(p_s)$, and transmission capacity $(p_b)$. The computing power is defined as the redundant computing power or redundant computing resources of the CPU, the storage capacity is the redundant storage space, and the transmission capacity is the redundant bandwidth. We calculate the energy requirements of the trust mechanism on a single node. The energy consumption sequence of node *i* is $EC_i = \{E(p_1), E(p_2), \ldots, E(p_m)\}$. The deployment note is *dnote*:

$$dnote = \begin{cases} 1, & if(E(p_1) < \lambda_1 p_1) \bigcap (E(p_2) < \lambda_2 p_2) \bigcap \cdots \bigcap (E(p_m) < \lambda_m p_m) \\ 0, & else, \end{cases} \tag{7}$$

where $\lambda_1, \lambda_2, \ldots, \lambda_m$ denote the adjustment coefficients. For example, the adjustment coefficient is equal to 0.8 if we hope the required capacity does not exceed 80% of the remaining capacity. For *i*-th node, each energy consumption factor is associated with some parameters, such as,

$$E(p_c) = f_i(d\_cal, r\_cal, ad\_cal), \tag{8}$$

$$E(p_s) = f_i(history\_vector, trust\_vector), \tag{9}$$

$$E(p_b) = f_i(hops), \tag{10}$$

where *d_cal*, *r_cal*, and *ad_cal* represent the energy consumption of direct trust value calculation, indirect trust value calculation, and trust value adjustment calculation, respectively. The *history_vector*, *trust_vector* are storage space consumption of the interaction history vector and trust value vector. The *hops* are the transmission distance of information. A simple function can be a summation, that is $E(p_c) = c \times (d\_cal + r\_cal + ad\_cal)$, $c$ is the unit energy consumption. The hops of indirect trust value calculation are related to calculation and transmission energy consumption.

Then we adjust the policy of deploying trust mechanisms on the network according to the above requirements.

*Case 1: static state*, when deploying a trust mechanism on the node.

*Step 1:* the trust management server traverses the network topology, records the node resource capacity.

*Step 2:* the server calculates the energy consumption of a standard trust calculation engine, which includes a trust computing and updating module (TM), a trust adjustment interface (TAI), and a trusted platform module (TPM).

*Step 3:* the server matches the energy according to formula (7) and decides whether each node is deployed a trust calculation engine or a partial engine (TM+TAI) or not.

*Case 2: dynamic runtime state*, the trust engine may need to be adjusted in real-time.

*Step 1:* the TAI monitors the change of the current node's residual energy and demanded energy consumption of the trust engine. Once $dnote = 0$, it records which resources do not match and reduce the strength of the trust engine.

*Step 2:* reduce corresponding $E(p_c)$, $E(p_s)$, or $E(p_b)$ so that $dnote = 1$.

It can reduce the hops of recommended trust calculation, do not calculate the recommended trust, turn off the TPM module to reduce $E(p_c)$ and $E(p_b)$. It can reduce the number of historical interaction sequences or trust value sequences, or request other redundant nodes to store them to reduce $E(p_s)$. It can reduce the communication frequency between the TAI and the trust server to reduce $E(p_b)$.

*2) The Delay:* The delay caused by the trust mechanism on a single device should not exceed the delay limit of an interactive process, and the delay in completing the task should be as small as possible.

Energy consumption and trust adjustment are closely related to time delay. Whether the adjustment of the trust calculation engine in a static state, or the adjustment of $E(p_c)$, $E(p_s)$, or $E(p_b)$ in a dynamic runtime state, the results are reducing the delay in or among the nodes. The delay in completing the task is the sum of the delay of a single node and the interaction and transmission between nodes.

Furthermore, the trust mechanism does not always increase the delay, it may also reduce the delay. For example, the trust engine can choose a trusted shortest path to transfer data to save time, the trust engine can detect and avoid malicious nodes from blocking and delaying the data.

*3) The Decision Impact on the Connectivity:* The trust decision-making method should ensure that the control process and key resources are available all the time. The connectivity evaluation needs to recognize the key resource nodes and the network nodes through which the control instructions flow.

Firstly, the trust management server receives the start signal from the instruction initiating node's TAI. Then it traverses the network and records the path set for transferring the instruction.

Secondly, the server monitors the network status. Once it receives the report of abnormal transmission from the nodes in the path, the server starts the adjustment in conjunction with the TAI of the relevant nodes.

*Case 1 (Insufficient energy consumption of key node cause transmission interruption):* The TAI of abnormal node adjusts according to *Case 2* in the previous section.

*Case 2 (The trust value of key nodes is less than the threshold. When there are redundant nodes, access requests should be shielded):* In a redundant network, there are multiple transmission paths for the control flow. Only nodes greater than the transmission threshold (i.e., 0.5) can participate in the operation flow. This mechanism isolates malicious nodes and prevents the system from being invaded. Traversing each path of the operation flow, the path through the trusted node is effective.

*Case 3 (No redundant nodes and untrustworthy critical nodes would cause transmission interruption of essential operations according to fine-grained access control):* The trust value of vital nodes of the critical operations is adjusted as follows:

$$T_R^* = T_R \bigcup Th_{R_t}, \tag{11}$$

where $T_R$ is the current trust value, $Th_{Rt}$ is the minimum trust threshold of the current operation (refer to fine-grained access threshold in formula (4)) and $T_R^*$ represents the adjusted trust value so that it meets the minimum trust threshold required for normal operation. The essential operations are not denied.

**Algorithm 1** Trust Calculation Algorithm

for $j = 1 : N$
1. The trust server determines the set of direct and indirect interactive nodes $Q_1$ and $Q_2$ of node $j$ according to the industrial network topology.
2. The trust server collects interaction information and operation behaviors related to node $j$ from $Q_1$ and $Q_2$.
3. For each interaction, the object evaluates the subject. If it is a successful interaction, the number of successful interactions increases by 1. Otherwise, the number of failed interactions increases by 1.
4. The node participating in the trust calculation collects the number of successful and failed transactions within the period $T$ or $N_{th}$, and calculates the direct trust value, indirect trust value, and global trust value of node $j$ according to formula (1) (2) (3).
5. Current or specified node stores the current trust value of node $j$ together with the historical trust record, recorded as $\{TV_1, TV_2, \ldots, TV_n\}$.
6. end

Then the trust server initiates the following measures for the abnormal node.

*Step 1:* If there is a standby node, the server compares the trust value of the standby node (the initial value is generally above the operation threshold), then switches to the standby node.

*Step 2:* If there is no standby node, the resident TAI shell or convert crucial data other than critical instructions into a non-operational mode, preventing the spread of attack information.

*Step 3:* The server starts an early warning mechanism to deliver the abnormal behavior to the intrusion detection module for further detection.

If necessary, we can adjust the threshold $Th_{Rt}$, expand the scope of trusted nodes to avoid key nodes being blocked.

*4) The Disposal and Recovery Time:* When the key nodes are short of energy, their trust values are lower than the threshold value, and there are no redundant nodes, and it suddenly encounters malicious attacks, the critical services will be interrupted. At this time, disposal and recovery are immediate. Reducing repair time can improve availability. When there is a standby node, the delay of switching from a failed / malicious node to a standby node is as small as possible. When there is an online detection and repair module, the restart and repair process is as short as possible.

In summary, the trust management server traverses through key control flow nodes and adjusts the trust decision mechanism under the availability constraints. On the one hand, it optimizes the trust scheme and reduces the deployment intensity. On the other hand, it optimizes the location of trust deployment. The node capabilities of the field network, control network, and process network of the industrial control system increase in sequence, and a more complex protection strategy can be deployed on the upper layer. Thirdly, it reduces the disposal and recovery time with the detection and recovery mechanism.

### *F. Summary of Trust Management Scheme*

We describe how the whole trust management scheme works in an industrial environment. The basic trust calculation algorithm for each node is given in Algorithm 1.

The trust management and adjustment algorithm are given in Algorithm 2.

**Algorithm 2** Trust Management Algorithm

1. We deploy the trust management server to the monitoring network bus, and deploy the trust engine to each node of the industrial control network.
2. The trust engine calculates the energy consumption of the trust calculation in Algorithm 1 according to (8)(9)(10), including the energy consumption of the static phase and the dynamic phase, and matches the energy consumption of the industrial network nodes according to (7). For unmatched nodes, the server takes the following measures in sequence until all nodes meet *dnote* $= 1$:
   1) Reduce the monitoring frequency of TPM;
   2) Reduce the communication frequency between TAI and server;
   3) Turn off the TPM function;
   4) Reduce the calculating range of recommended trust (*i.e.*, the hop of recommended nodes decreases from 2 to 1);
   5) Only consider the direct trust value;
   6) Turn off the TM function.
3. The trust engine starts the monitoring function. Once it finds an abnormality of the interactive node, it determines the type, content, and affected object of the abnormality through Table II. Then the trust engine determines the threat level according to the method in Section IV.C, and adjusts the current trust value of the target node.
4. The trust engine determines the trusted threshold of various operations. The server determines the dynamic operation authority of each node in the network according to the fine-grained access control method in (4).
5. The server identifies key nodes and paths and selects the shortest trusted path to transmit instructions.
6. When the path or the key node is blocked, the trust engine adjusts the trusted state of the key node according to (11) to make it connectable, and perform disposal and recovery at the same time.
7. The server judge whether the delay of completing the task meets the requirements, if not, follow steps 2 to further simplify the location and strength of the trust deployment until the delay requirements are met.

## V. Trust Management Deployment Scheme

The trust management deployment scheme is shown in Fig. 3. The proposed trust management is mainly deployed on the industrial network, that is, between the field device and the DMZ. The trust management server is deployed on the bus of the process control network. The management functions include the centralized collecting of trust information, centralized storage and scheduling of trust information, coordination of trust calculation and updating, and reporting abnormal information to the intrusion detection system. The trust server adopts a mainstream configuration and general interface.

The trust calculation engine is deployed in each calculating node's redundant space. The trust calculation engine includes three configurable parts: a trust computing and updating module (TM), a trust policy adjustment interface (TAI), and a trusted platform module (TPM). TM is responsible for the trust collection of neighbor nodes, trust calculation and updating, and trust-based decision-making. TAI interacts with the trust management server. TAI realizes energy monitoring (which needs to interact with the operating system), upload and download trust data, start and stop of trust mechanism. It adjusts the deployment and calculation strategy of trust based on the requirements of individualized trusted decisions and availability constraints, such as whether the field device activates a trust calculation engine and how to adjust the complexity of indirect trust calculation. All of these functions are realized in software mode. TM and TAI are customized software codes (also the functional code modules) that can be embedded into the application software of the target node. The TPM can be deployed on the engineer workstation, data server, an OPC server, etc., to avoid unauthorized attacks on the server's operating system,

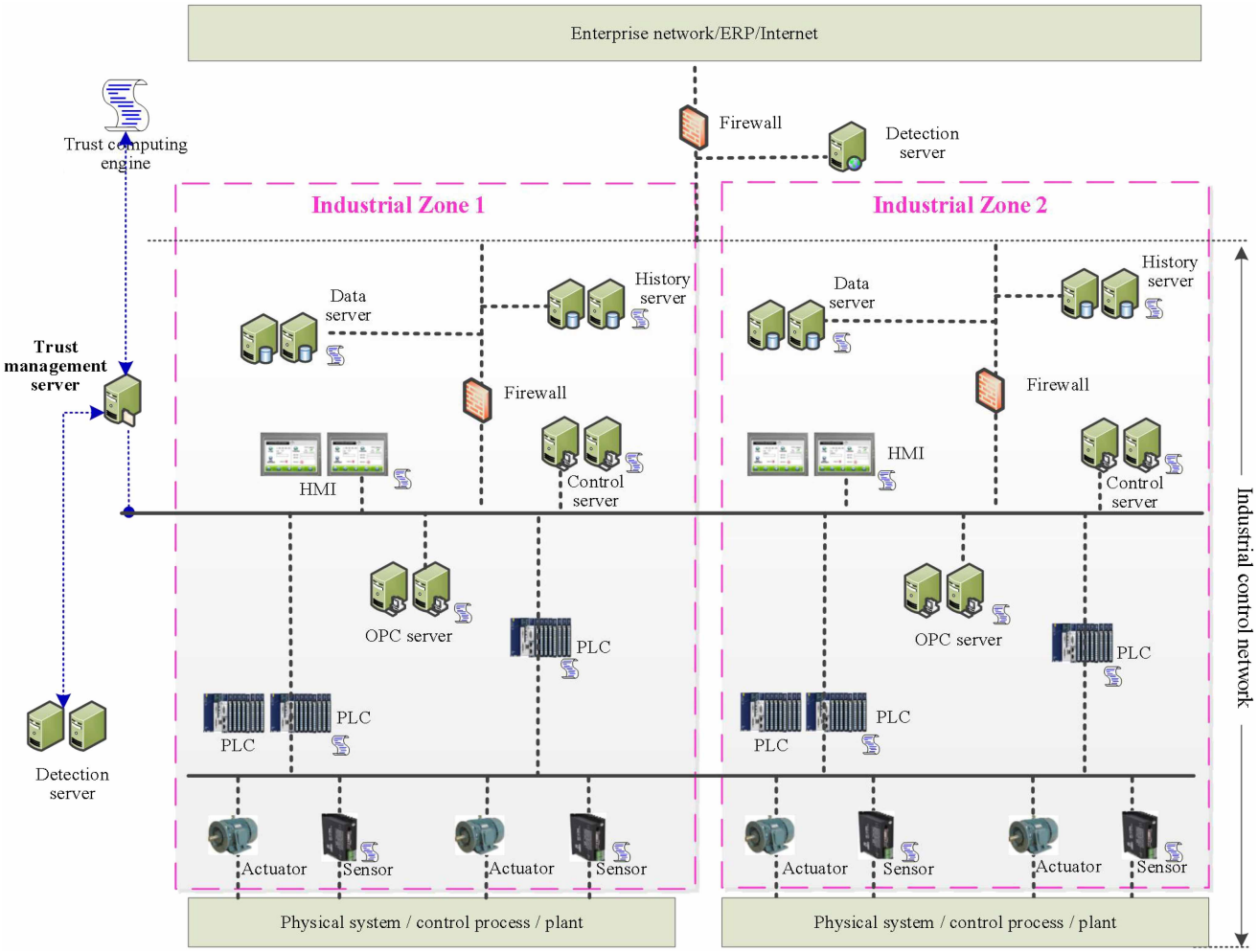



Fig. 3. Trust management deployment scheme.

data leakage, and unauthorized access to the OPC server from peripherals. TPM is a commercial hardware-supported software system, which needs the support of the hardware PCIe board of the target node and matching trusted management software.

The trust engine and the connection with the operating system are shown in Fig. 4. The green icon is realized by programmable software code, and the dark gray box is guaranteed by hardware and memory space.

The target node should meet some configurations when it is deployed the trust engine: it has redundant storage space, which is used to store the implementation code of the trust engine, trust value sequence, and interactive historical sequence; it should support the interactive interface between the trust engine and the operating system, the trust engine can affect the related operation of the operating system based on trust decision; it has trust transmission port, which can share the Ethernet port with data transmission.

The database server and history server are assumed by the general server, which can provide adequate memory. We can load the trust engine on their operating system, call the trust engine during communication, and assist in operation decision-making; HMI, control server, and OPC server are special servers. HMI runs large-scale man-machine interface software to display monitoring screens of the operation status in real-time. The control server adopts WinCC configuration software to support the configuration of the control equipment. OPC server supports OPC classic protocol and OPC DA communication. These dedicated servers adopt mainstream configuration, and their running capacity and memory are more powerful than general-purpose servers. We can also load the trust engine on the operating system (Windows, Linux) and call the trust engine to assist in operation decision-making; PLC and sensor are special equipment, and their calculation and storage capacity varies among different products. For example, Siemens S7-200 PLC does not support software configuration. The upper computer software can load some functions of the trust engine (*e.g.*, the trust decision code) into the S7-300 / 400 PLC through the serial port. S7-1500 has larger memory and supports complex trust functions. However, due to the closeness of the controller, it does not support TPM, nor does it have a reserved PCIe board or USB interface. Ordinary sensors do not support software configuration, and intelligent sensors support network tools to configure their App, and the trust calculation code can be loaded, but the TPM is not supported. The software implementation of TPM

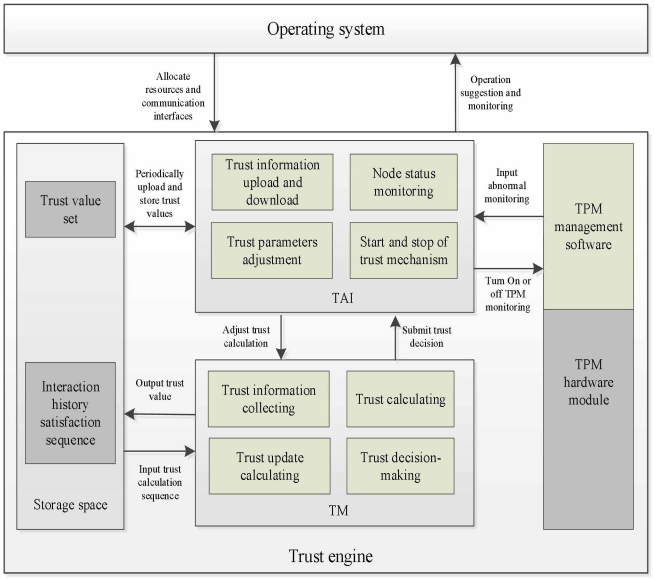



Fig. 4. Structure and interface of the trust engine.

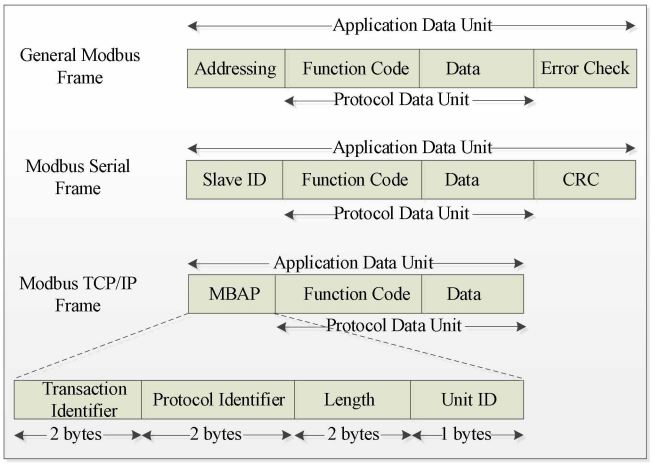



Fig. 5. MODBUS frame structure [3].

is an idea, but it is not discussed in this paper. The Firewall supports OPC, Modbus, S7, and other common industrial protocols. Therefore, in the industrial control network shown in Fig. 3, the trust engine can be deployed in the data server, history server, HMI host, control server, OPC server, some PLCs, and some intelligent sensors.

The intrusion detection/defense system deployed on the firewall side of the enterprise network mainly detects and blocks partial attacks from the external network. The detection system deployed in the monitoring network discovers the network's abnormal behavior and provides it to the trust server or trust engine for trust modeling. When the trust engine finds an exception or an important node fails, it will be sent to the intrusion detection module for further processing in addition to trust adjustment.

The industrial control system adopts a hierarchical model. Suppose the industrial process is a power scenario, such as a nuclear power generation system. The key functions include reactor power control, pressurizer pressure control, steam generation and emission control, turbine control, etc. These functions are completed by a single or multi-level DCS network. The parameters of turbine control include power, frequency, pressure, speed, flow, etc. Each industrial process can be implemented in different industrial areas. As shown in Fig. 3, industrial zone 1 assumes the control of reactor power, and zone 2 is a pressure adjustment process. It should be noted that, in addition to the two functions of direct control and monitoring control in one domain, the industrial control system can also carry out cross-domain cooperation control. For example, when the control server of the neighborhood fails, the engineer workstation can coordinate the control server in industrial zone 1 to monitor the PLC in zone 2. In process control, when the control process of industrial zone 1 is completed, the control server in zone 1 and zone 2 should hand over and transfer the control process to zone 2 to avoid errors.

The industrial control system uses the bus to connect various industrial equipment, and data is exchanged between equipment through a special protocol. The commonly used industrial protocols include Modbus, Foundation Fieldbus H1 / HSE, PROFIBUS / PROFINET, etc. Modbus protocol can work on Fieldbus in serial mode or on the control network and monitoring network in Ethernet architecture and is compatible with TCP protocol. Modbus protocol works in the data link layer (MAC layer), and the physical layer can support RS-232, RS-485 electrical interfaces. The structure of the data frame is shown in Fig. 5.

Application data unit (ADU) of general Modbus frame includes addressing, function code, data, and error check. The addressing indicates the address of the slave. The function code indicates the action to be performed, and the slave responds to the action and returns the same function code. The data area indicates the data information to be performed or collected, including the actual value, data register address, etc. Error checks generally adopt hash values or CRC verification. The Modbus serial frame is generally used by the control server or HMI connecting to the field controller and I/O device through the serial gateway. The *Slave ID* is the identity of the slave device. The master device addresses the slave device and initiates a request. In the Modbus TCP / IP frame, the address domain is replaced by the header of the Modbus application protocol (MBAP). The information includes the transaction identifier (associated TCP transaction), the protocol identifier, the remaining field size indication (length), and the host identification. The error verification is completed by TCP / IP. The complete Modbus ADU is encapsulated in the data field of the standard TCP / IP frame.

OPC server is a software interface or standard driver based on Microsoft's OLE / COM technology. The HMI and control server can use the OPC client and OPC DA/HDA communication to access and exchange real-time data and historical data. DMZ and its upper network nodes use traditional TCP / IP protocol to transmit data.

The data frame transmitted by TCP / IP protocol and Modbus protocol includes a source address, destination address, execution action, data and its location. All data frames

conforming to the specification belong to normal frames or authorized frames. The destination node will process the data and return messages if necessary.

The current node extracts the source address of the data frame, and the resident trust engine will judge whether the source node is trusted according to the trust status of the source node, the data access control requirements of the current node, and the results of intrusion detection (if any), and block or warn the data frame of the untrusted node. The trust engine extracts the actions in the data frame and rejects the actions that do not meet the access regulations or unauthorized actions. The trust engine judges the credibility of the resident node. When the node is abnormal, the trust engine prevents the sending or shelling of data frames to avoid threat diffusion. According to the next target address extracted by the data frame, the trust engine judges the trust status of the target node, and assists process the instruction data to be sent, such as selecting the trusted path node (changing the destination address and identity in the data frame) to transmit data.

In summary, after the trust mechanism is deployed, it will work in the manner mentioned in the previous Section IV. The trust server initializes the trust model according to the network topology, possible control procedures, and data. Then the trust engines collect the interaction operation behaviors between various components in the industrial control system, including the processing, transmission, and storage of data and instructions. Then the engines calculate the direct trust value of the interacted nodes and assist other nodes in calculating the recommended trust value. TPM and TAI in the trust engine identify the abnormal behavior in the interaction process, determine the threat degree, and assist TM to update the trust value. Meanwhile, TAI monitors the energy consumption and connectivity of nodes, adjusts the trust decision mechanism according to the method in the previous section.

The availability constraints mainly exist in the control network and field area. The node capabilities of the field network, control network, and process network increase in sequence and a more complex protection strategy can be deployed on the device in the upper layer.

## VI. Analysis and Simulation Experiment

### A. Scheme Characteristics Analysis

The proposed trust management scheme has the following features:

The trust management scheme follows the fundamental processes of the trust mechanism, such as trust generation and modeling, trust information collection, trust calculation and update, and trust decision-making. The trust modeling process follows the best practices. Trust modeling considers direct trust, recommended trust, reward and punishment mechanisms, and the adjustment calculation according to the number of interactions, which conforms to the primary attributes of trust in human society. The time attribute of weight reflects flexibility. Direct trust adopts a subjective logic method, which embodies the unity of the trust's subjective identification and objective evaluation. Indirect trust adopts the nodes' recommendation within the domain. Considering the nodes' task relevance and real-time requirements of the industrial network, it is reasonable to consider nodes' recommendations within 2 hops. The trust is routinely updated according to the time window under normal conditions. Under abnormal conditions, the scheme considers the difference between abnormal operations and adopts a punitive update method, it conforms to the trust law in human society. The trust decision-making considers three typical scenarios: optimal path selection, fine-grained access control, and anomaly detection of non-redundant nodes. The entire trust modeling process is in the same line as other distributed scenarios. Meanwhile, the trust mechanism is expected to be applied to industrial control networks.

The trust management program aims to identify and deal with the abnormal operation of the control instructions. By analyzing the unknown threats represented by "Stuxnet" and "Ukraine's power outages," it is clear that the key to the attack lies in the tampering, blocking, and malicious feedback of control commands on industrial network nodes. The control command's abnormal operation behavior is concerned goal of our trust management. In the modeling process, the scheme considers multi-dimensional malicious behavior identification and difference modeling of essential operational behavior. The different importance of operation is reflected in the punishment factor embedded in the trust update process. Compared with the existed trust methods, our model's accuracy in industrial scenarios is improved.

To solve trust management's availability problem in industrial control network scenarios, the trust management mechanism adopts lightweight modeling algorithms, flexible update mechanisms, and hierarchical deployment schemes. Trust modeling uses linear weighting and subjective logic as the primary method to minimize time delay. Trust update and decision-making adopt an adjustable mechanism. We reduce the recommendation scope and diminish the complexity of recommended trust on nodes with high real-time requirements. When the trust value of critical nodes is lower than the threshold, trust compensation and audit response methods ensure network connectivity. Based on the suggestion from the policy adjustment interface, the trust servers push the differential trust calculation method to the trust engine of the key nodes. According to the different availability requirements of components at each level of the industrial control system, we arrange complicated trust management servers, complete trust engines on the monitoring network, and install a simple trust calculation engine on PLC, which uses limited redundant resources to collect and calculate trust. The deployed method ensures the continuity of the industrial control network's operations.

### B. Active Immune Analysis

*1) Effectiveness Evaluation of Trust Modeling:* Increasing nodes' trust value with good performance and reducing nodes' trust value with abnormal behavior can realize the converge cooperation between nodes. The evaluation effect of trust refers to three parameters: recognition accuracy ($RA$), trust sensitivity ($TS$), and processing rationality ($PR$), that is, $Evaluation = \{RA, TS, PR\}$.

*RA:* It is reflected in the process of trust modeling. The operation behavior is expressed statistically, and abnormal behavior is regarded as failure interaction. The abnormal type is recorded and confirmed by the trust engine. The trust value considers the multi-hop neighbor recommendation and the reward and punishment based on statistical behaviors. Abnormal behavior changes should be consistent with changes in the trust relationship.

*TS:* It is reflected in the difference in abnormal behavior recognition. The proposed method establishes a quantitative relationship between different abnormal behaviors, positions, operation modes, and threat levels in trust updating. The proposed method reflects these differences in punishment factors so that the difference in abnormal behavior is reflected in the change rate of the trust value.

*PR:* It is reflected in trust-based decision-making. The path optimization process will shield untrusted nodes. Fine-grained access control determines access rights according to the range of trust values, and access by untrusted nodes or nodes with lower trust values will be restricted, thus maximizing the security of transmission instructions.

*2) Active Immunity:* The core effect of active immunity includes two aspects: normal operations by normal behaviors nodes are efficient and undisturbed, abnormal operations cannot affect the system. For normal operations, it satisfies:

$$p\_status(n,t) = 1, \forall n \in N_R, \forall t \in T, \tag{12}$$

where *p_status* is the execution status, it is normal at any node and any time. And for abnormal operations, it satisfies:

$$p\_status(n,t) = 0, \exists n \in N_R, \exists t \in T, \tag{13}$$

the malicious behavior must be recognized and handled at a certain node or moment.

*Remark 1:* the proposed trust management method makes the target system active immune.

For normal operations, normal behavior nodes generally have the trust value range required for normal operation authority and can continuously process and transmit data and instructions at any time. The trust decision adjustment based on availability constraints described in Section IV-E can ensure the connectivity of normal key operations. Therefore, it can operate normally at any node.

For abnormal operations, there are three types of abnormal behavior referred to Table II.

*Unauthorized operation of control instructions:* Unauthorized operations may occur on equipment on the instruction transmission link. The trust engine can actively identify abnormal behaviors, the TPM will warn of violations of normal operations on the devices and boundaries, and the TAI combined with the trust server can identify abnormalities at the network and its boundary. According to the mutual evaluation, the trust engine can collect each node's behavior evaluation sequence, and calculate the nodes' trust value. The trust value of abnormal behavior nodes will decline or accelerate the descent. According to fine-grained access control rules, untrusted nodes cannot perform/access related operations/resources, at least with restricted access; when there are redundant nodes, untrusted nodes cannot participate in the transmission of the critical instructions. Therefore, unauthorized operations are blocked at a certain node or moment.

*Non-compliant operation of control instruction:* It includes operations, *i.e.*, modification, deception, discarding control instructions by the authorized nodes, the mis-operation by legal operators. As malicious internal entities generally do it, the malicious behavior is easier to detect and record by TPM, and the trust value is calculated. The trust-based decision-making process is similar to unauthorized operations. Because the non-compliance behavior directly interacts with the control instruction, it is more severe, and the punishment factor is larger than the unauthorized action. Once it happens, its trust value drops faster, and it will likely fall below the trust threshold after 1-2 abnormalities.

*Interfere with regular operation:* The devices may block information transmission, or send a large number of packets to the same address, etc. These behaviors disrupt normal business. Abnormal behavior nodes can be shielded based on the trust isolation mechanism and access control mechanism in the proposed method. The unexpected behavior fails to be responded to at a certain node. The engine activates alerting and diagnosing and minimizing the spread of abnormal behavior.

It notes that the trust management mechanism belongs to a trusted third party. It needs to cooperate with the detection server, audit module for online or offline processing to realize the active immunity of the network system.

*3) Trade-Off of the Trust Adjustment and Protection Effect:* The trust adjustment based on availability constraints may reduce the strength of trust deployment and the accuracy of trust evaluation.

*Remark 2:* optimized adjusted strategy makes the new vulnerability failure at least not expand.

*Note 1 (We would adjust the deployment location and strength of the trust engine):* In theory, all nodes in the industrial control network are deployed with a trust engine. Due to availability constraints, field devices or some devices in the control network may not deploy a trust engine, or not deploy a TPM, or only deploy a communication verification mechanism. Because these devices are controlled equipment, the risk of being attacked can be controlled under the conditions that the control behavior and operation data of the main control equipment are effectively protected.

*Note 2 (We would adjust the trust computing strength):* Reducing the computing strength of recommendation trust can improve availability. Because the historical trust sequence and direct trust value are the dominant factors of trust calculation, there are few redundant paths in the industrial control network, so the reduction of calculating indirect trust value has a limited impact on the results. The detecting and audit module can reduce the risk caused by the decrease in the trust update process.

*Note 3 (We would adjust the trust threshold to ensure connectivity):* We may temporarily increase the trust value of untrusted key nodes, which threatens the surrounding nodes. Since the untrusted node has been monitored or processed

(e.g., switching to standby node, switching to non-operational mode, warning), the threat is controllable.

### C. Availability Analysis

Availability is reflected in running continuity and increased delay. The delay in deploying the trust solution does not affect the industrial control system's regular operation as much as possible, and the trust management process does not block the normal operation flow.

*Remark 3:* the trust mechanism can optimize or maintain the normal operation flow.

*Note 1 [Trust mechanism can identify the above third type of abnormal behavior (interfere with regular operation) and avoid abnormal blocking of operation]:* The trust adjustment mechanism also ensures the connectivity of normal operation flow.

*Note 2 (The trust model considers the characteristics of physical information fusion):* The physical device failure or physical attack will also produce abnormal behaviors. Abnormal behavior identification of information side and physical side both are considered in trust management.

*Lemma 1:* The time and space complexity of deploying the trust mechanism is controllable. The time complexity is $O(n^2)$, the space complexity is $O(n)$.

The time complexity of trust calculation and update of the proposed scheme is $O(n_1^2)$, $n_1$ is the number of nodes involved in collecting, calculating, and updating trust information, which is generally within the hops range of the neighbor, the number is controllable. According to the relationship between hops and delay, it is reasonable to calculate the indirect trust value within two jumps. The computational complexity of the capability matching algorithm is $O(n_2^2)$, and $n_2$ is the number of nodes where the trust management scheme is to be deployed. Therefore, the proposed trust management scheme's time complexity on a single node is calculated: $O(n_1^2) + O(n_2^2)$. The number of nodes of the trust engine and the number of nodes calculated by the matching algorithm are related to the network scale. Therefore, the protection scheme's time complexity is $2O(n^2)$, $n$ is the average node number with linear complexities, and the number is controllable. Assuming that the number of network nodes is 100, a single node completes the calculation at *us* level with the computing power ($10^6$ *ips*) of the CPU of the control device, which is acceptable in most industrial control scenarios. Information transfer includes the transfer time of collected trust information, the query and response time of indirect information, and the storage time of trust information. The time complexity of transferring trust information is $O(n_3)$, $n_3$ is the number of transmission nodes, and the longest path is multi-hop distances from the trust management server to the furthest trust engine. The number is controllable. Suppose the processing time unit is $t_1$ and the 1-hop distance delay is $t_2$, then the overall delay is $t_1 * 2O(n^2) + t_2 * O(n)$. According to the business's allowable delay, it can be determined whether the current node's delay meets the requirements. The policy of calculation or transmission needs to be adjusted through the policy adjustment interface if it is not. It should be noted that under abnormal conditions, the delay of the inspection and audit is not considered.

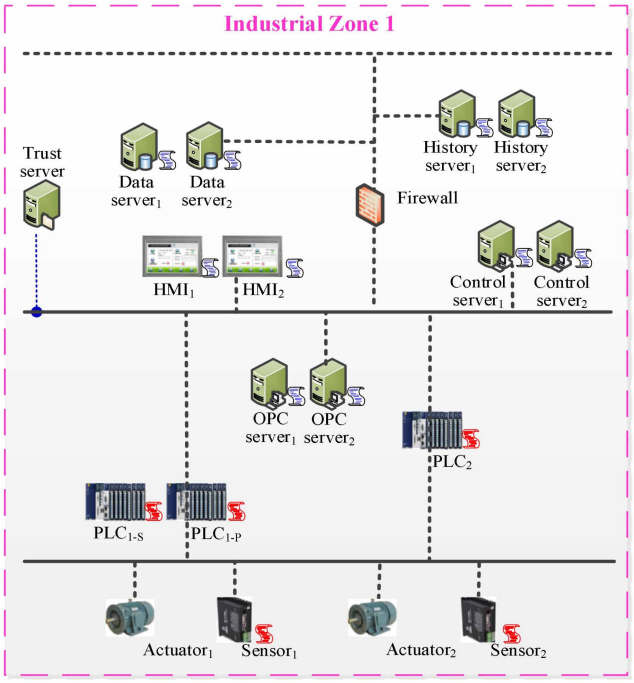



Fig. 6. Trust engine deployment in each zone.

The space complexity of the trust scheme: the space complexity mainly increases the trust management-related server and the corresponding data storage (the behavior sequence of several-hop neighbors and trust vector). The increase in memory is linear, and the complexity is $O(n)$. The storage resource added is little for the trust scheme.

### D. Simulation Experiment

The proposed trust management solution is deployed refer to Fig. 3. We deploy four industrial zones, and the deployment method in each zone is shown in Fig. 6. Assuming that the control process is aimed at the process industry, cross-domain data flow and monitoring control are allowed between the four industrial zones. Each industrial zone has 15 bidirectional communications nodes, including two data servers, two history servers, two HMI, two control servers, two OPC servers, three PLCs including one standby PLC, and two sensors. Therefore, the total number of network nodes in the four zones is 60. The firewall does not participate in trust calculation, and the actuator does not have a bidirectional communication function. We deploy the trust engine on each bidirectional communication node. PLC and sensor are resource-sensitive nodes (shown by the red icon in the figure), there may be no redundant resources to deploy the trust engine. We assume that under static conditions, $PLC_{1-P}$ and $Sensor_2$ cannot be deployed the trust engine, and all PLCs cannot be deployed the TPM. And in the running state, we assume $PLC_2$ in two zones cannot be deployed to the trust engine. Therefore, the number of nodes with the complete trust engine is 40, and the number of nodes with the partial trust engine is 50. The communication protocol between PLC and sensor is Modbus

TABLE V
SETTINGS OF SIMULATION PARAMETERS

| Parameters | $N_1$ | $N_2$ | $T_0$ | $\alpha$ | $\beta$ | $N_d$ | $m$ | $\lambda$ |
|---|---|---|---|---|---|---|---|---|
| Value | 60 | 40~50 | 0.5 | 0.7 | 0.3 | 15 | 3 | 0.8 |

Serial protocol, and Modbus TCP protocol is used for communication between the control server, HMI, and PLC. TCP / IP protocol is adopted between the external network and the data server. The time delay of the control process is required to be within 50*ms*, and connectivity is highly required. It is not allowed to interrupt the control process.

The simulation parameters are shown in Table V. $N_1$ represents the number of nodes in the industrial control network. $N_2$ represents the number of nodes deploying trust mechanisms. Trust deployment is dynamic, and the number is about 40-50. $T_0$ is the initial trust value, $\alpha_0(t)$ and $\beta_0(t)$ are the initial weights of the direct trust and the indirect trust calculation, generally within a period *T*, the proportion is unchanged. In the simulation, it is assumed that the weight $\alpha = \alpha_0(t)$, $\beta = \beta_0(t)$. $N_d(t)$ is the threshold of the number of direct interactions. It is a fixed value of $N_d = 15$. Beyond this number, the trust engine only calculates the direct trust value, and $\alpha$ and $\beta$ are no longer applicable. *m* is the number of capacity measures. We consider three typical capacities of computing, storage, and transmission. And the adjustment coefficient uses the same value $\lambda$.

To take the abnormal operation of control instructions as an example, we simulate some typical abnormal behavior:

1) Simulate the unauthorized read and write operations of the control instructions of the industrial control networks. The control instructions are transferred among the data servers, HMI, control servers, PLCs, etc. We construct a control instruction and transmit it through the data server, HMI, control server, PLC, etc., and simulate operations, e.g., reading, writing, communication, processing, etc. This is a legal operation. Then we access an unauthorized device in the network and perform operations, e.g., reading, writing, and requesting communication on the constructed control instruction. These are abnormal behavior. We describe a schematic diagram of abnormal behavior in Fig. 7. The purple line and blue line represent the flow chart of the established control instructions. The first control instruction shown in the purple line passes through the $Data\ server_2$, $HMI_1$, $Control\ server_1$, $PLC_{1-S}$, and $Sensor_1$. Assuming that $Data\ server_1$ and $HMI_2$ are unauthorized nodes, they may read, write and request services for the control instruction.

2) Simulate the non-compliant read and write operations on the data server, HMI, control server, PLC, sensor, etc., of the industrial control networks. We construct a control command, load it to the data server, HMI, control server, PLC, etc., and simulate reading, writing, communication, processing, etc., these are legal operations. We then load authorized devices, randomly read, delete, and insert control instructions, not transmitting information, transmitting false information, and discarding instructions. These are abnormal behavior. As shown in Fig. 7, authorized nodes $HMI_1$ and $PLC_{1-P}$ may initiate non-compliant operations, such as malicious deletion, insertion, discarding, and the false transmission of the control instructions.

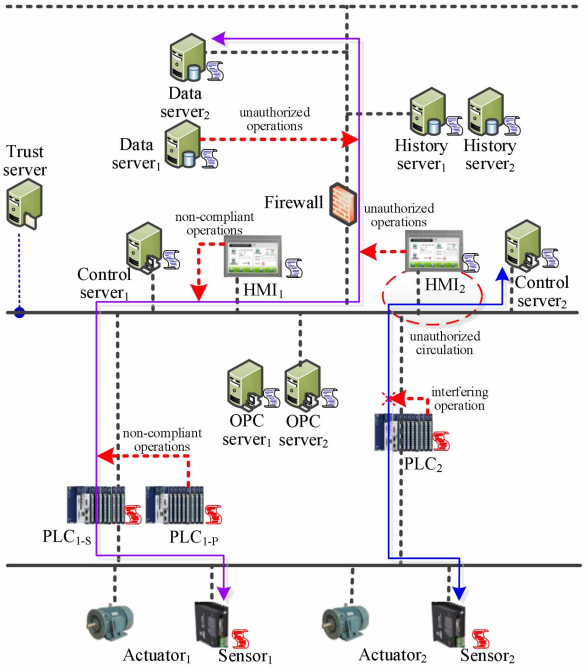


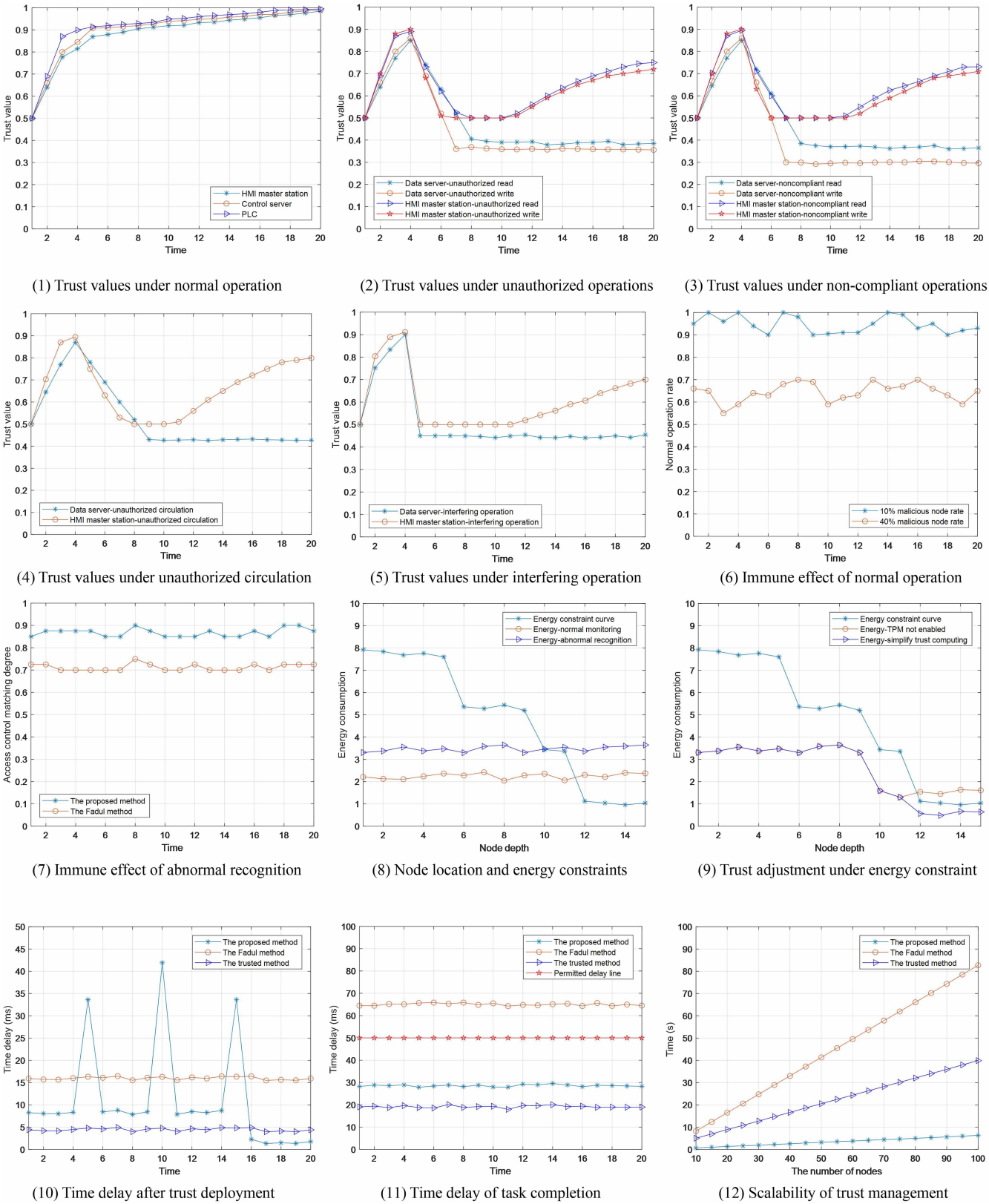

Fig. 7. A schematic diagram of abnormal behavior.

3) Simulate the control commands circulating among unauthorized nodes. We construct a program code as a control command and transmit it through the data server, HMI, and control server through an open port. Some nodes (e.g., HMI) on the path are unauthorized, and they cooperate with malicious nodes to deliver the data, which can be regarded as unauthorized circulation. Assuming that $HMI_2$ is an unauthorized node, the second control instruction shown in the blue line in Fig. 7 passes through the unauthorized node.

4) Simulate the abnormal behavior of interfering with regular operations. We construct a program code as a control command and transmit it through the data server, HMI, and control server through an open port. Then we select a node that randomly blocks information transmission at different times. These behaviors disrupt normal operations. Assuming that $PLC_2$ fails or is attacked, the normal flow of the second control instruction in Fig. 7 will be blocked.

Based on the defined scenarios and parameters, we perform 12 sets of simulations, and the results are shown in Fig. 8.

Fig. 8 (1) depicts the change of trust values under regular operation. The abscissa is time (number of interactions), and the ordinate is the node's trust value. The comparison experiments are the change curves of the trust value of selected typical three key nodes (HMI operation master station, control server, and PLC). As the number of normal operations increases, the successful feedback gradually increases. According to the trust calculation formula, the trust value will gradually increase. It can be seen from the figure that the trust value gradually increases under normal operation until it approaches 1.




(1) Trust values under normal operation

(2) Trust values under unauthorized operations

(3) Trust values under non-compliant operations

(4) Trust values under unauthorized circulation

(5) Trust values under interfering operation

(6) Immune effect of normal operation

(7) Immune effect of abnormal recognition

(8) Node location and energy constraints

(9) Trust adjustment under energy constraint

(10) Time delay after trust deployment

(11) Time delay of task completion

(12) Scalability of trust management

Fig. 8. Simulation results.

Fig. 8 (2) depicts the trust changes of nodes under unauthorized reading and writing operations on the control flow. We simulate unauthorized reading and writing operations at the fifth interaction and return to normal operation at the tenth interaction. The abscissa is the number of interactions, the ordinate is the node's trust value, and the comparison

experiments are the change curves of the trust values of selected typical two nodes: data server, and HMI operation master station (short for HMI master station). We set that the data server is a common node of data flow, and the HMI station is the key node of instruction transmission. It can be seen the trust value of key nodes drops significantly during abnormal behavior, the HMI station is more important than the data server, and reading and writing operations have different penalties, as shown in Table IV. The trust value of the HMI station decreases faster than that of the data server, and the trust value of the write operation decreases more quickly than that of the read operation. A vital node's trust value is above the threshold (0.5) to maintain the normal control flow at this node (or replacement node) according to the trust adjustment mechanism. Non-important nodes with trust values less than the threshold are directly shielded. Therefore, it loses the next interaction opportunity. After the 11th interaction, the trust value of a vital node (well-behaved node or replacement node) is slowly recovered, and the trust value of the write operation recovers slower than that of the read operation. The trust value declines fast and recovers gradually, it is in line with the trust law in human society.

Fig. 8 (3) depicts the trust changes of the node under non-compliant reading and writing operations on the control flow. We simulate non-compliant reading and writing operations at the fifth interaction and return to normal operation at the tenth interaction. The abscissa is the number of interactions, the ordinate is the trust value of a node, and the comparison experiments are the change curves of the trust values of selected typical two nodes (data server, HMI master station). It can be seen that the results are similar to those in Fig. 8 (2). The penalty for non-compliant operations is greater than unauthorized operations, under the same distance from the core PLC, refer to Table IV. And the trust value of the write operation decreases more quickly but recovers slower than that of the read operation. Therefore, the trust value drops faster and recovers more slowly.

Fig. 8 (4) depicts the trust changes of the node on the control link under the control instruction flowing through unauthorized nodes. We simulate unauthorized transmission operations at the fifth interaction and return to normal circulation at the tenth interaction. The abscissa is the number of interactions, the ordinate is the trust value of a node, and the comparison experiments are the change curves of the trust values of selected typical two nodes (data server, HMI master station). It can also be seen that the results are similar to those in Fig. 8 (2). The penalty for unauthorized circulation is lower than that for unauthorized operations; therefore, the vital node's trust value drops more slowly and recovers faster than that of the corresponding node in Fig. 8 (2). Non-important nodes with trust values less than the threshold (0.5) are directly shielded, and the trust value no longer increases.

Fig. 8 (5) depicts the trust changes of the node on the control link under the abnormal behavior of interfering with regular operation. We simulate interfering with regular operation at the fifth interaction and recovering at the tenth interaction. The abscissa is the number of interactions, the ordinate is the trust value of a node, and the comparison experiments are the change curves of the trust values of the data server and HMI master station. The interception of control instructions is the most serious case. The trust engine directly reduces the trust value of the data server below the threshold, isolates, and repairs non-important nodes. For critical nodes, the trust engineer reduces its trust value to 0.5 and repairs it immediately under the premise of ensuring connectivity. The trust value starts to recover at the 12th time.

Fig. 8 (6) describes the immune effect of the trust model. The abscissa is the number of interactions, and the ordinate is the normal operation rate, which is defined as the ratio of the times where the instruction is operated without interference under 100 operation tasks. Task interference is induced by unavailable services due to a low trust value (the read operation requires a trust value of 0.5 or above), malicious blocking, and hops exceeding the count limit. The specific malicious nodes and malicious behaviors in each interaction are randomly assigned. We can see from the figure that under 10% of malicious nodes, the normal operation rate is above 90%. As long as there are no severe malicious behaviors in the backbone nodes, it can operate successfully; under 40% of malicious nodes, most of the normal operation rate exceeded 60%. On some times, because malicious nodes occupy the core path of the network, there is no trusted path, resulting in more failures.

Fig. 8 (7) describes the immune effect of abnormal behavior recognition. Effective recognition and interruption of abnormal behavior is another important aspect of active immunity. The abscissa is the number of interactions, and the ordinate is the access control matching degree, which is defined as the proportion of correct access control times to total access control times under 20 operations. According to formula (4), different trust values correspond to different access control permissions. Abnormal behavior adopts 20% non-compliance operation. It is consistent with the working rules of malicious behavior in Fig. 8 (3). The nodes with abnormal behavior are randomly assigned each time. The comparison algorithm is the Fadul scheme [30], which adopts reputation-based management in a communication-based backup protection system to mitigate network vulnerabilities in smart grid devices. Trust calculation also includes the calculation of direct trust, recommended trust, and global trust value. The access control is based on a fixed threshold. We can see from the figure that the proposed method has higher accuracy in blocking malicious behavior compared with the Fadul scheme. The Fadul method lacks differential trust modeling and punishment mechanism for malicious behavior.

Furthermore, we verify the impact of the proposed algorithm on the availability of industrial control networks. As analyzed previously, availability is mainly related to energy consumption, time delay, repair time, etc.

According to the industrial control network in Fig. 3, we first set the energy consumption limit of the nodes. Then we number the nodes from top to bottom. The depth of the first node closest to the external network is 1. The depth is extended to the field layer in sequence and is increased by 1 for each hop. The nodes with the deepest hops are the field nodes with limited resources. For example, one hop corresponds to one

node for simplicity, nodes 1-5 belong to DMZ and its upper network, nodes 6-9 belong to the process network, nodes 10-11 belong to the control network, and nodes 12-15 belong to the field area.

Fig. 8(8) depicts the relationship between the redundant energy of nodes and the energy consumption of deploying a complete trust mechanism on the same nodes. The abscissa is the depth of the nodes, and the ordinate is the energy consumption curve. The energy consumption of calculation is far greater than that of storage and transmission, we only consider the former. The energy consumption of calculation is the multiple of the proportion of the unit energy consumption of calculation related to the corresponding total energy (e.g., the remaining capacity of CPU). The reference energy curve is 80% of the redundant energy ($\lambda = 0.8$). The actual energy consumption of calculation is calculated by the formula (8). It can be seen from the figure that the energy consumption of some field layer nodes exceeds the warning value, which is not suitable for deploying a complete trust mechanism in the normal monitoring stage. While in the abnormal recognition stage, some control network nodes are also not suitable for deploying a complete trust mechanism.

Fig. 8(9) describes the deployment adjustment of the trust mechanism based on Fig. 8(8) in the abnormal recognition stage. First, we turn off the TPM function in the control network nodes and reduce the hops of trust value calculation from 2 to 1 in the field nodes. The curve in the figure shows that the field layer nodes still do not meet the energy requirements. Further, we only calculate the direct trust value in the field nodes, so that the energy consumption of the field layer nodes can meet the energy limitation.

Fig. 8 (10) describes the time delay of a single node after deploying the trust mechanism. The abscissa is the number of interactions, and the ordinate is the time to complete the interaction, which is the sum of the necessary transmission and processing time, trust calculation time, and the attack identification time. The comparison methods are common reputation algorithms - the Fadul method [30] and the trusted computing scheme - Ref. [28]. The trusted method applies trusted computing to the industrial control systems, it identified the node based on the trusted hardware. It can authenticate the neighbor nodes by a distributed trusted discovery protocol based on a provable security method, and transmit instructions through trusted nodes. This protocol can be used to monitor, detect, and locate attacks. The interaction time based on the trusted scheme is the sum of required transmission and processing time and identity authentication time. And the interaction time based on the behavioral reputation method is the sum of required transmission and processing time and calculation time of behavior credibility. We can see from the figure that in the routine monitoring stage, the proposed method's time delay is between [28] and [30]. When at the point of the abnormal recognition stage (*i.e.*, the 5th, 10th, 15th interactions), the time delay is relatively high. When the number of interactions is enough, *i.e.*, the number of interactions between two nodes reaches the threshold value $N_d$ (after the 15th interaction in the figure), the trust engine only considers the direct trust value of each other, and the calculation complexity will reduce significantly.

Fig. 8 (11) describes the time delay of task completion after deploying the trust mechanism. We adjust the trust deployment method according to Fig. 8(9). The abnormal monitoring and task transmission are processed in parallel in the current node. Or we request the upper node or trust server to process abnormal recognition. The abscissa is the number of transactions. The proportion of malicious nodes is 10%. In each transaction, the task transmission path is randomly assigned. The ordinate is the time delay, which is defined as the maximum transmission delay among different paths. We adopt the same method to count the delay in the compared the Fadul method and the trusted scheme. Generally speaking, the acceptable task delay of the industrial control system is within 50 *ms*. We can see from the figure that the time delay of the proposed method and the trusted computing method are within the permitted delay line. Because the Fadul method has no trust adjustment mechanism, the delay does not meet the requirements.

Fig. 8 (12) describes the network scalability with the trust mechanism. The abscissa is the number of network nodes, the initial number is 10, and increases by 5 each period. The nodes number in different network levels increases in the same proportion. The total number is 100. The ordinate is the time to complete the trust management of all the nodes from a server perspective, which is the sum of the transmission and processing time, network and trust adjustment time, and the abnormal identification and confirmation time. The comparison methods are the Fadul method [30] and the trusted computing scheme [28], the interaction time based on the trusted scheme is the sum of required transmission and processing time, identity authentication, and confirmation time. And the interaction time based on the reputation method is the sum of required transmission and processing time, and management and adjustment time of behavior credibility of all the nodes. With the increase in the number of nodes, the time of trust management is increasing. Because the proposed algorithm allocates the work of the server and trust engine, the server is only responsible for routine management, and its time increases linearly. The time of the trusted scheme increases proportionally with the increase of nodes, and the time of the reputation scheme presents a rapid increase. Compared with the Fadul method and the trusted scheme, the time of the proposed algorithm increases slowly, which proves that it has better scalability.

In summary, the industrial control network's trust model has better immunity to abnormal behavior of the control flow and can be deployed in an industrial control system with availability constraints.

## VII. Summary

### A. Discussion

There are three types of decision-making mechanisms based on trust management, corresponding to three types of application scope.

Path optimization and malicious nodes shielding based on trust values are applicable in distributed redundant networks, that is, in large-scale distributed control systems, networked

control systems on the industrial field, industrial cloud, and edges networks in the cyber-physical systems. It is not applicable in a fixed or single-point network (such as a PLC control system).

The fine-grained access control based on trust values is mainly used to authenticate the node access or evaluate differentiated threats. When an external node accesses the industrial control system, it is necessary to verify the credibility of its identity, judge its trustworthiness, and determine whether it is allowed to access based on the sensitivity of the resource. In the differentiated threat assessment, different behaviors correspond to different threat degrees. A simple example is that the threat of a malicious write operation to control instructions is greater than that of a read operation, so a write operation is expected to be operated by more trusted nodes. Fine-grained access control should be made for different behaviors. However, fine-grained access control needs more time. How to apply it to a strong real-time environment needs further investigation.

The linkage between trust and detection and audit applies to various scenarios. The trust engine submits the untrusted behaviors to the detection or audit module. Detection and audit servers can audit whether there is an intrusion in real-time or semi-real-time, and recover when necessary.

### *B. Summary and Outlook*

This paper proposed a trust management model for industrial control networks oriented to the recognition and management of abnormal behaviors. The proposed trust model performs trust modeling, calculation, and decision-making against abnormal behaviors operating on the control flow under the influence of unknown threats. It can protect control instructions against abnormal operation, and realize the immune effects collaborated with the trust engine (including a TM, TPM, and TAI or a part of them), detection, and audit mechanism. The trust management scheme is adjusted according to the availability constraint function of entity capability, operation delay, and the connectivity of key nodes to ensure the normal transmission of the industrial control system's control flow. Analysis and simulation results show that the proposed scheme has better performance and controllable complexity.

This work is mainly the identification and credibility management of abnormal behaviors in the typical industrial scenarios described in Section V. Due to the versatility and the semantic-level characteristics of the trust mechanism, this method can also be considered for security protection for other similar industrial networks. This article only focuses on the unified abnormal behavior on the physical side and the information side. If we can mine more fine-grained abnormal behavior and its classification from the cyber-physical fusion characteristics of the industrial control system, we will improve the accuracy of security protection. This article is based on simulation, the testing of the trust management's deployment scheme in an actual control system needs to be further realized.

## REFERENCES


[1] M. Cheminod, L. Durante, and A. Valenzano, "Review of security issues in industrial networks," *IEEE Trans. Ind. Informat.*, vol. 9, no. 1, pp. 277–293, Feb. 2013.

[2] K. O. Akpinar and I. Ozcelik, "Methodology to determine the device-level periodicity for anomaly detection in ethercat-based industrial control networks," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 2, pp. 2308–2319, Jun. 2021.

[3] D. Pliatsios, P. Sarigiannidis, T. Lagkas, and A. G. Sarigiannidis, "A survey on SCADA systems: Secure protocols, incidents, threats and tactics," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 1942–1976, 3rd Quart., 2020.

[4] *Recommended Practice: Improving Industrial Control Systems Cybersecurity With Defense-in-depth Strategies*, United States Dept. Homeland Security, Washingdon, DC, USA, Oct. 2009.

[5] A. A. Jillepalli, F. T. Sheldon, D. C. de Leon, M. Haney, and R. K. Abercrombie, "Security management of cyber physical control systems using NIST SP 800-82r2," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, 2017, pp. 1864–1870.

[6] U. Darshana and S. Sampalli, "SCADA (supervisory control and data acquisition) systems: Vulnerability assessment and security recommendations," *Comput. Security*, vol. 89, Feb. 2020, Art. no. 101666.

[7] L. Zhou, K.-H. Yeh, G. Hancke, Z. Liu, and C. Su, "Security and privacy for the Industrial Internet of Things: An overview of approaches to safeguarding endpoints," *IEEE Signal Process. Mag.*, vol. 35, no. 5, pp. 76–87, Sep. 2018.

[8] G. Drosatos, K. Rantos, D. Karampatzakis, T. Lagkas, and P. Sarigiannidis, "Privacy-preserving solutions in the Industrial Internet of Things," in *Proc. 16th Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, 2020, pp. 219–226.

[9] D. Serpanos, "Secure and resilient industrial control systems," *IEEE Des. Test*, vol. 35, no. 1, pp. 90–94, Feb. 2018.

[10] S. Huang, C.-J. Zhou, S.-H. Yang, and Y.-Q. Qin, "Cyber-physical system security for networked industrial processes," *Int. J. Autom. Comput.*, vol. 12, no. 6, pp. 567–578, 2015.

[11] Z. Zhang, R. Deng, D. K. Y. Yau, P. Cheng, and J. Chen, "Analysis of moving target defense against false data injection attacks on power grid," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 2320–2335, 2020.

[12] C. Shen *et al.*, "Research on trusted computing and its development," *Sci. China Inf. Sci.*, vol. 53, no. 3, pp. 405–433, 2010.

[13] Y. Wang, G. Cui, L. Zhang, and H. Li, "Research on application of trusted computing 3.0 in industrial control system of nuclear power plant," in *Proc. 12th Int. Conf. Commun. Softw. Netw. (ICCSN)*, 2020, pp. 297–301.

[14] H. Hu, J. Wu, Z. Wang, and G. Cheng, "MIMIC defense: A designed-in cybersecurity defense framework," *IET Inf. Security*, vol. 12, no. 3, pp. 226–237, 2018.

[15] Y. Wang and J. Vassileva, "Bayesian network-based trust model," in *Proc. IEEE/WIC Int. Conf. Web Intell. (WI)*, 2003, pp. 372–378.

[16] N. Sardana, R. Cohen, J. Zhang, and S. Chen, "A Bayesian multiagent trust model for social networks," *IEEE Trans. Comput. Soc. Syst.*, vol. 5, no. 4, pp. 995–1008, Dec. 2018.

[17] Y. L. Sun, W. Yu, Z. Han, and K. J. R. Liu, "Information theoretic framework of trust modeling and evaluation for ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 305–317, Feb. 2006.

[18] H. Jadidoleslamy, M. R. Aref, and H. Bahramgiri, "A fuzzy fully distributed trust management system in wireless sensor networks," *AEU Int. J. Electron. Commun.*, vol. 70, no. 1, pp. 40–49, 2016.

[19] D. Velusamy, G. Pugalendhi, and K. Ramasamy, "A cross-layer trust evaluation protocol for secured routing in communication network of smart grid," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 1, pp. 193–204, Jan. 2020.

[20] G. Theodorakopoulos and J. S. Baras, "On trust models and trust evaluation metrics for ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 318–328, Feb. 2006.

[21] G. Han, J. Du, C. Lin, H. Wu, and M. Guizani, "An energy-balanced trust cloud migration scheme for underwater acoustic sensor networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1636–1649, Mar. 2020.

[22] N. Somu, M. R. G. Raman, K. Krithivasan, and V. S. S. Sriram, "A trust centric optimal service ranking approach for cloud service selection," *Future Gener. Comput. Syst.*, vol. 86, pp. 234–252, Sep. 2018.

[23] X. Li, J. Yuan, E. Li, W. Yao, and J. Du, "Trust-aware and fast resource matchmaking for personalized collaboration cloud service," *IEEE Trans. Netw. Service Manag.*, vol. 16, no. 3, pp. 1240–1254, Sep. 2019.

[24] N. Li, V. Varadharajan, and S. Nepal, "Context-aware trust management system for IoT applications with multiple domains," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, 2019, pp. 1138–1148.
[25] H. Okhravi and D. M. Nicol, "Application of trusted network technology to industrial control networks," *Int. J. Crit. Infrastruct. Protect.*, vol. 2, no. 3, pp. 84–94, 2009.
[26] S. Pinto, T. Gomes, J. Pereira, J. Cabral, and A. Tavares, "IIoTEED: An enhanced, trusted execution environment for Industrial IoT edge devices," *IEEE Internet Comput.*, vol. 21, no. 1, pp. 40–47, Jan./Feb. 2017.
[27] O. A. Harshe, N. T. Chiluvuri, C. D. Patterson, and W. T. Baumann, "Design and implementation of a security framework for industrial control systems," in *Proc. Int. Conf. Ind. Instrum. Control (ICIC)*, 2015, pp. 127–132.
[28] N. Göttert, N. Kuntze, C. Rudolph, and K. F. Wahid, "Trusted neighborhood discovery in critical infrastructures," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, 2014, pp. 976–981.
[29] J. Fadul, K. Hopkinson, C. Sheffield, J. Moore, and T. Andel, "Trust management and security in the future communication-based 'smart' electric power grid," in *Proc. 44th Hawaii Int. Conf. Syst. Sci.*, 2011, pp. 1–10.
[30] J. E. Fadul, K. M. Hopkinson, T. R. Andel, and C. A. Sheffield, "A trust-management toolkit for smart-grid protection systems," *IEEE Trans. Power Del.*, vol. 29, no. 4, pp. 1768–1779, Aug. 2014.
[31] W. Zeng and M. Chow, "A reputation-based secure distributed control methodology in D-NCS," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6294–6303, Nov. 2014.
[32] C. Boudagdigue, A. Benslimane, A. Kobbane, and J. Liu, "Trust management in Industrial Internet of Things," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3667–3682, 2020.
[33] C. Marche and M. Nitti, "Trust-related attacks and their detection: A trust management model for the social IoT," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 3, pp. 3297–3308, Sep. 2021.
[34] W. Fang, M. Xu, C. Zhu, W. Han, W. Zhang, and J. J. P. C. Rodrigues, "FETMS: Fast and efficient trust management scheme for information-centric networking in Internet of Things," *IEEE Access*, vol. 7, pp. 13476–13485, 2019.
[35] I. U. Din, M. Guizani, B. Kim, S. Hassan, and M. K. Khan, "Trust management techniques for the Internet of Things: A survey," *IEEE Access*, vol. 7, pp. 29763–29787, 2019.
[36] M. Al-Khafajiy *et al.*, "COMITMENT: A fog computing trust management approach," *J. Parallel Distrib. Comput.*, vol. 137, no. 1, pp. 1–16, 2020.
[37] A. Jøsang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decis. Support Syst.*, vol. 43, no. 2, pp. 618–644, 2007.
[38] Y. He, C. Liang, F. R. Yu, and Z. Han, "Trust-based social networks with computing, caching and communications: A deep reinforcement learning approach," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 1, pp. 66–79, Jan.–Mar. 2020.
[39] Y. Wang, I.-R. Chen, J.-H. Cho, and J. J. P. Tsai, "Trust-based task assignment with multiobjective optimization in service-oriented ad hoc networks," *IEEE Trans. Netw. Service Manag.*, vol. 14, no. 1, pp. 217–232, Mar. 2017.
[40] I.-R. Chen, J. Guo, D.-C. Wang, J. J. P. Tsai, H. Al-Hamadi, and I. You, "Trust-based service management for mobile cloud IoT systems," *IEEE Trans. Netw. Service Manag.*, vol. 16, no. 1, pp. 246–263, Mar. 2019.
[41] S. Huang, Z. Zeng, K. Ota, M. Dong, T. Wang, and N. N. Xiong, "An intelligent collaboration trust interconnections system for mobile information control in ubiquitous 5G networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 1, pp. 347–365, Jan.–Mar. 2021.
[42] S. Huang, A. Liu, S. Zhang, T. Wang, and N. Xiong, "BD-VTE: A novel baseline data based verifiable trust evaluation scheme for smart network systems," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 3, pp. 2087–2105, Jul.–Sep. 2021, doi: 10.1109/TNSE.2020.3014455.
[43] Y. Liu, A. Liu, X. Liu, and M. Ma, "A trust-based active detection for cyber-physical security in industrial environments," *IEEE Trans. Ind. Informat.*, vol. 15, no. 12, pp. 6593–6603, Dec. 2019.
[44] W. Mo, T. Wang, S. Zhang, and J. Zhang, "An active and verifiable trust evaluation approach for edge computing," *J. Cloud Comput.*, vol. 9, no. 1, p. 51, Sep. 2020.
[45] E. K. Wang, Y. Li, Y. Ye, S. M. Yiu, and L. C. K. Hui, "A dynamic trust framework for opportunistic mobile social networks," *IEEE Trans. Netw. Service Manag.*, vol. 15, no. 1, pp. 319–329, Mar. 2018.
[46] *Security for Industrial Automation and Control Systems Part 1: Terminology, Concepts, Models*, ANSI/ISA Standard 99.00.01, 2007.
[47] J. Pawlick and Q. Zhu, "Strategic trust in cloud-enabled cyber-physical systems with an application to glucose control," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 12, pp. 2906–2919, Dec. 2017.
[48] A. Jøsang, "A logic for uncertain probabilities," *Int. J. Uncertainty Fuzziness Knowl. Syst.*, vol. 9, no. 3, pp. 279–311, Jun. 2001.
[49] N. B. Akhuseyinoglu, M. Karimi, M. Abdelhakim, and P. Krishnamurthy, "On automated trust computation in IoT with multiple attributes and subjective logic," in *Proc. IEEE 45th Conf. Local Comput. Netw. (LCN)*, 2020, pp. 267–278.
[50] N. Milanovic and B. Milic, "Automatic generation of service availability models," *IEEE Trans. Services Comput.*, vol. 4, no. 1, pp. 56–69, Jan.–Mar. 2011.

**Jingpei Wang** received the bachelor's and master's degrees from China Three Gorges University, Yichang, China, in 2007 and 2010, respectively, and the Ph.D. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2014. Then he joined the China CEPREI Laboratory in 2014 and became a Senior Engineer in 2017. He joined Zhejiang University in 2018. He is currently a Research Fellow with the School of Control Science and Engineering, Zhejiang University, Hangzhou, China. His research interests include network security, ICS information security, and trust management.

**Zhenyong Zhang** received the bachelor's degree from Central South University, Changsha, China, in 2015, and the Ph.D. degree from Zhejiang University, Hangzhou, China, in 2020. He was a Visiting Scholar with the Singapore University of Technology and Design, Singapore, from 2018 to 2019. He is currently a Professor with the College of Computer Science and Technology, Guizhou University, Guiyang, China. His research interests include cyber–physical system security, applied cryptography, and machine learning security. Dr. Zhang serves as the Review Editor of Frontiers in Communications and Networks and reviewers for IEEE TRANSACTIONS ON PLASMA SCIENCE, IEEE TRANSACTIONS ON SMART GRID, IEEE TRANSACTIONS ON AUTOMATIC CONTROL, IEEE TRANSACTIONS ON CONTROL OF NETWORK SYSTEMS, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He also served as the Session Chair for ASCC 2022.

**Mufeng Wang** received the Ph.D. degree in control science and engineering from the South China University of Technology, Guangzhou, China, in 2019. He was a Research Fellow of Control Science and Engineering with Zhejiang University, Hangzhou, China, from 2019 to 2021. He is currently a Senior Engineer with the China Industrial Control Systems Cyber Emergency Response Team. His research interests include analysis and synthesis of security for industrial control systems and cyber–physical systems.