IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 22, 2018, accepted November 7, 2018, date of publication November 12, 2018, date of current version March 20, 2019.

Digital Object Identifier 10.1109/ACCESS.2018.2880838

Trust Management Techniques for the Internet of Things: A Survey

IKRAM UD DIN[1], (Senior Member, IEEE), MOHSEN GUIZANI[2], (Fellow, IEEE), BYUNG-SEO KIM[3], (Senior Member, IEEE), SUHAIDI HASSAN[4], (Senior Member, IEEE), AND MUHAMMAD KHURRAM KHAN[5], (Senior Member, IEEE)

[1]Department of Information Technology, The University of Haripur, Haripur 22620, Pakistan
[2]College of Engineering, Qatar University, Doha 2713, Qatar
[3]Department of Software and Communications Engineering, Hongik University, Sejong Campus, Sejong 2639, South Korea
[4]InterNetWorks Research Laboratory, School of Computing, Universiti Utara Malaysia, Sintok 06010, Malaysia
[5]Center of Excellence in Information Assurance, King Saud University, Riyadh 11451, Saudi Arabia

Corresponding author: Byung-Seo Kim (jsnbs@hongik.ac.kr)

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2018-0-01411, A Micro-Service IoTWare Framework Technology Development for Ultra small IoT Device).

ABSTRACT A vision of the future Internet is introduced in such a fashion that various computing devices are connected together to form a network called Internet of Things (IoT). This network will generate massive data that may be leveraged for entertainment, security, and most importantly user trust. Yet, trust is an imperative obstruction that may hinder the IoT growth and even delay the substantial squeeze of a number of applications. In this survey, an extensive analysis of trust management techniques along with their pros and cons is presented in a different context. In comparison with other surveys, the goal is to provide a systematic description of the most relevant trust management techniques to help researchers understand that how various systems fit together to bring preferred functionalities without examining different standards. Besides, the lessons learned are presented, and the views are argued regarding the primary goal trust which is likely to play in the future Internet.

INDEX TERMS Internet of Things, trust management techniques, trust contributions, trust limitations.

I. INTRODUCTION

Internet of Things (IoT) is a new model developed to allow millions of smart communication nodes be connected to the Internet [1]. Such nodes are sensors and/or actuators that can process and retrieve data from other devices with or without human interference [2]. The IoT development brings a remarkable effect to various areas, such as smart cities [3], smart healthcare [4], smart transportation [5], cellular communications [6], data mining [7], manufacturing [8], and environmental monitoring [9] among others [10]–[13]. This high level of heterogeneity, linked with the IoT system, is presumed to increase security threats for the existing Internet, which is used to let humans interact with machines [14]. Conventional privacy solutions and security provisions do not satisfy user requirements because of their limited processing power.

In an IoT environment, a variety of independent devices cooperate with one another to perform different tasks. These devices in such dense environment irrationally discover other devices. Such discovery is named as semantic discovery which creates different information trust related issues [15]. Various methods to achieve semantic interoperability include broker based architecture and different service platforms. Broker architecture is complex and weak to handle the object-to-object discovery [16]. Trust management in an IoT environment is provided by various methods [17], [18] which employs past experience, sensor data irregularity, reliability, and availability as trust matrices.

Different than other networking environments, IoT faces new challenges due to its particular features. The most crucial of these features, apart from privacy and security, is trust. That is, if the aggregated information from different devices is malicious and not sufficiently trustworthy, it is difficult to be accepted by users albeit the trust of application layer and network layer are fully provided [19]. The most important question that arises is to know that how the IoT generated data is converted into useful information to provide a secure and trustworthy communication. For this reason,

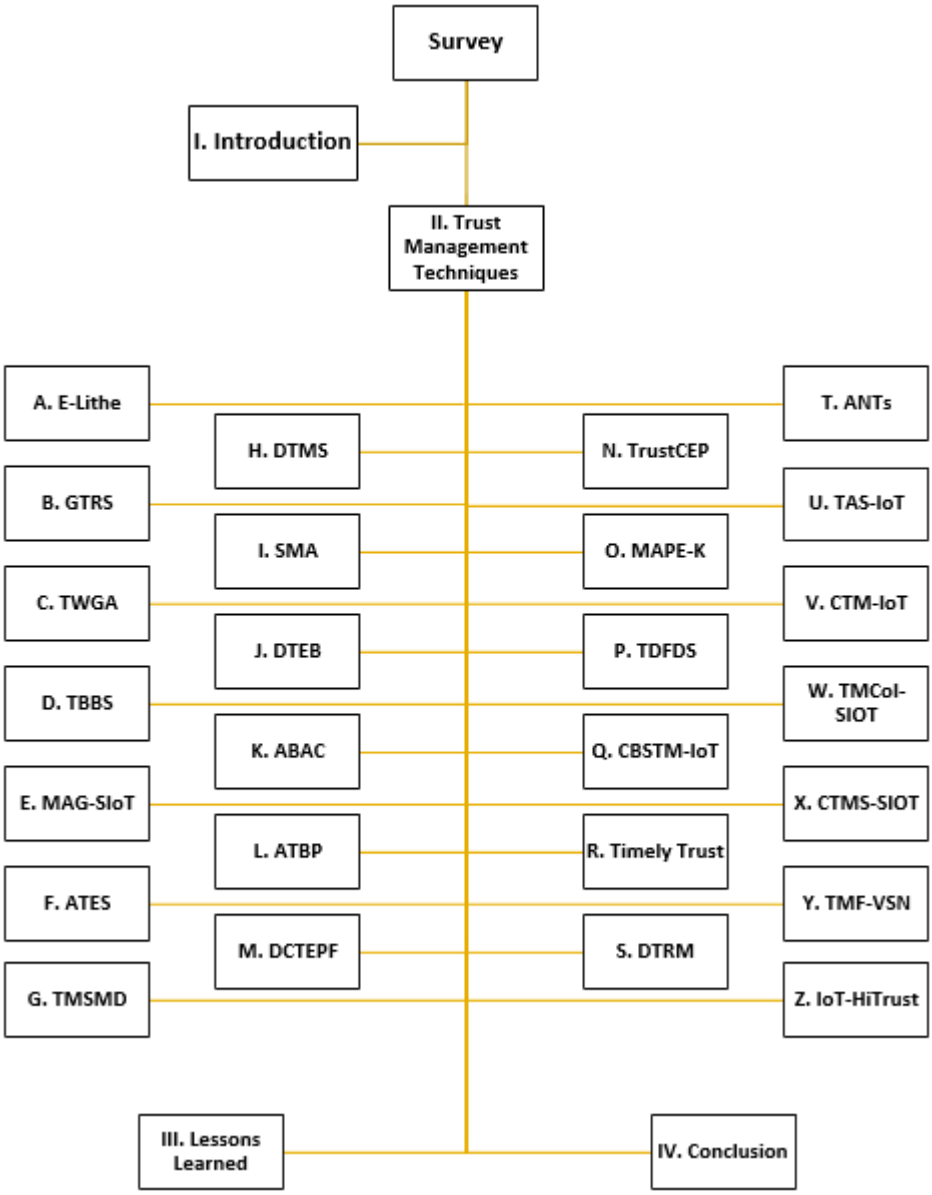

FIGURE 1. Structure of the survey.

various techniques have been developed, several of which are discussed in this paper in detail with respect to their system models, contributions, and limitations.

A few papers have surveyed the IoT trust management, e.g., [14] and [19], but these surveys provide a general discussion on IoT trust and do not discuss the available trust management models. To the best of our knowledge, we are the first to provide a comprehensive survey of IoT trust management techniques.

The organization of this survey is such that Section II presents different IoT trust management techniques, Section III discusses the learned lessons, and Section IV concludes the survey. The overall structure of this paper is depicted in Figure 1 in a very simplified style.

II. TRUST MANAGEMENT TECHNIQUES

IoT is an emerging technology that provides a base to replace the traditional communication systems with a modern one [20]. In this system, machines perform different operations to handle changing situations in real life without the involvement of human efforts. IoT permits nodes (things) to have different characteristics and share services and information [21]. *Things* produce decentralized networks with adaptable topologies. In this specific situation, it is essential to have a strong, versatile and reliable communication, and correspondence among these devices [21]. Various devices, for example, computers and mobile phones, work together to make humans' life more comfortable. With this rising number of connected devices, it is hard to assume that which device

is trustworthy [10], [22]. For this reason, several approaches have been developed, which are presented in this section comprehensively.

A. E-LITHE

IoT is a new emerging paradigm that connects millions of things (devices) through the Internet. However, the connection of this huge number of devices needs a secure communication. To provide secure communication within the IoT environment, the idea of Datagram Transport Layer Security (DTLS) [23] is adopted to construct Transport Layer Security (TLS) over datagram [24]. DTLS is a protocol which allows secure communication between client-server applications over the Internet in a secure manner. It is based on the TLS that provides prevention of message forgery, tempering, and fragmentation. It also deals with the packet re-ordering, loss of datagram, and size of the datagram. However, DTLS is defenseless for Denial of Service (DoS) attacks and needs thousands of computations for constrained devices. A DoS attack is a type of attack that disrupts network services and thereby prevents a successful communication between two devices. In the DoS attack, the device availability is targeted by the attacker and the requested services are not provided to the legitimate user [25].

To overcome DTLS' shortcomings for constrained devices, the proposed work focuses on an enhanced and lightweight DTLS, namely Enhanced Lightweight DTLS for IoT (E-Lithe) [26]. The concept of Trusted Third Party (TTP) is added to provide enhancement to E-Lithe. TTP provides pre-exchanging of secret keys as well as resilience against DoS attacks. In the proposed model, the Next Header Compression (NHC) and the IP Header Compression (IPHC) are used as compression schemes.

Figure 2 illustrates the enhanced handshake protocol for E-Lithe. First, before starting the actual handshaking phase, the server and the TTP protocol agree on sharing a secret key for successful communications. In the next step, the mutual key is shared between the TTP protocol and the client. The sharing of a mutual key between the client and the TTP protocol prevents extra burden of energy consumption on the server and authenticates the client-server communication. In the client-server communication, a client sends a handshake to the server with its authentication key. If the authentication key matches, the server validates the process of ''Hello'' message. If the key does not match, the process of Hello message is terminated.

To ensure the lightweight transmission for constrained devices, the E-Lithe adopts a compression strategy, which reduces the power consumption as well as prevents the overload of fragmentation. The compression strategy is comprised of record layer, handshake layer, and client Hello. The record layer is further composed of version, epoch, sequence number, and fragment. The handshake layer consists of message type and message sequence. Excluding length details, the message type and the sequence are sent in the actual state. In the client Hello message packet, the first four bits are set as identity bits and the last four bits represent the session ID, cookie, cipher suite, and the compression mechanism.

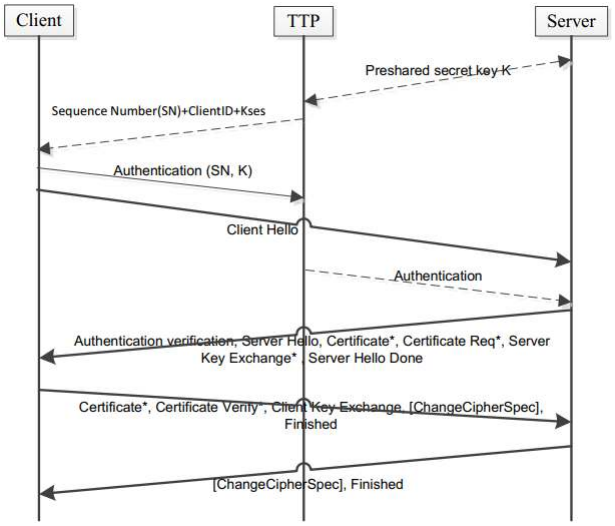

FIGURE 2. Communication in the E-Lithe scheme is subject to four rules: i) before the handshaking phase starts, the server and TTP approve a pre-shared secret key; ii) the client and TTP share that key for secure client-server communications; iii) the client requests the server with a handshake message and the server validates that if its authentication key matches; iv) upon authentication, the server sends Hello message to the client, otherwise, terminates the session [26].

The E-Lithe scheme enhances security for constrained devices by adding the concept of TTP, which in turn decreases DoS attacks by sharing secret keys. The cookie exchange technique in the E-Lithe scheme provides more efficiency and reduces computational overhead in comparison with the Lithe [24] and DTLS [23] schemes. However, if an intruder creates multiple handshake requests from multiple nodes, then the battery drainage is a crucial problem to handle frequent computations.

B. GTRS

Recommender Systems (RS) have a high popularity to predict and suggest items based on past ratings [27]. An RS may be of three types: content-based filtering (CBF), collaborative filtering (CF), and a hybrid system. In the IoT environment, billions of devices are interconnected with each other, where each device requires and provides services being a part of an IoT network. All devices are discovered, which provide a certain type of service. After discovering the service, the next step is the selection of service from the list of available services. The major issue in this scenario is the service selection, which may lead to the confusion of trusting other devices. The existing central CF recommender lacks two key performance measures. First, it saves the rating matrix in memory to predict the best service from the memory. Second, it leads to data sparsity issues.

To address these issues, a scalable algorithm is proposed, which is based on the CF recommendation. To overcome the non-competency of the central RS, the proposed

recommender predicts the best-rated service provider (SP) and selects it to retrieve services from that SP. Based on the idea of [28], the Graph-based Trust-enhanced Recommender System (GTRS) [27] adopts the concept of Social IoT (SIoT) and forms a new social relationship between nodes in the IoT network. In the context of trust among devices, the use of recommendations of friends and friends-of-friends can be dealt with.

In this model, a requesting node sends a request to its friends to get recommendations for the past ratings. If it finds the best-rated node, then it selects that particular node to get services from it. Otherwise, the request is forwarded to the friends-of-friends. The trust among nodes is calculated in two ways, i.e., direct social trust where nodes are connected directly, and indirect social trust in which nodes are indirectly connected. The Indirect trust can be calculated using trust propagation and aggregation. The effect of a node can be calculated using two types of methods, i.e., trust and similarity. Trust and similarity can be calculated from ratings and network structures. In the context of trust calculation from ratings, the GTRS adopts the O'Donovan and Smyth [29] approach to calculate the correlational trust among nodes. Only one predictor is used to calculate the predicted ratings for a requesting node.

Trust can also be calculated using network structures where nodes are free to develop a relationship with each other, the same way as with humans. In the IoT environment, a device can make a relationship of four types: i) The co-owner relationship that occurs when nodes are owned by the same owner and its trust level is 4; ii) the friendship relationship occurs when owners of nodes are friends and its trust level is 3; iii) the co-location relationship occurs when nodes are at the same location and its trust level is 2; and iv) the co-parental relationship is given a trust level 1 and it occurs when the manufacturer of nodes is the same. Trust may also be calculated by combining centrality and trust level. Centrality represents that how one node is central to the other. Hence, trust can be measured by the combination of both parameters.

In the proposed system, each node is capable of calculating its own predictions for the best rated services. In addition, it computes the effectiveness of one node on another by combining their trust and similarity. However, the proposed recommender is not able to predict the rating for a device if somebody has not rated it. Moreover, it is difficult to tackle the prediction issues when searching nodes are similar to each other.

C. TWGA

A trustworthy gateway architecture (TWGA) [30] for the IoT environment is proposed against malicious attacks, such as spoofing and DoS, without the involvement of a heavyweight individual security technique. The existing trust models are based on individual device security techniques and logical addressing. The proposed architecture is compatible with the existing system, which renovates the IP address of IoT devices and uses the control server as an Identifier (ID).

The TWGA architecture consists of the following components and their functions:

1) Initially, the path is established among trust domains through the ID-path setup function. Both trust domains send a device-ID, signature, and public key to each other for establishing a trusted-ID between a home device and an SP. The virtual IP, their ID verification, and the public and private keys are stored in a cached-ID table.
2) After configuration of the path, data forwarding packets along with signatures are transmitted to a smart home gateway domain (SHGD) against the path-ID via a forwarding function. For example, before forwarding the ID-packet to the SHGD, the virtual IP address of a device is transformed into a destination address-ID. The SHGD, after verification, finds the source and destination IP addresses with the help of a cached-ID table and then forwards the data packet to a device when required.
3) For the authentication and verification of packets, private/public keys are used to verify whether the sent ID-packet is correct or not.
4) The ID-packet engine provides a route for data packets, while Domain Name System (DNS) converts these IDs into IP addresses or vice versa. The cached and registered ID tables are repositories that store information regarding private/public keys, IDs, and IP addresses.

Besides, secure key remains the same if an intruder gains the private key by any means. That is, the intruder can inject false data and make a repudiation attack.

D. TBBS

At the time of any emergency or accident, information sharing is a time sensitive matter and requires fast responses. Wrong emergency information may cause several distinct problems on roads. Therefore, a secure mechanism is needed to show if a vehicle is trustworthy and reliable or not. In addition, it is needed to know that other vehicles can rely on the shared data of a particular vehicle. To avoid these issues, a trust and behavior-based system (TBBS) is proposed in the IoT enabled vehicular ad-hoc network (VANET) [31]. The proposed mechanism is based on behavior and trust, where the traveling information and behavior of vehicles are monitored by the base transceiver signal station (BTSS). The proposed Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication system is depicted in Figure 3.

In the TBBS, it is mandatory that all vehicles must be equipped with collision detection sensors for the convenient deployment of airbags. A vehicle must have a transceiver to share information among various vehicles as well as with the BTSS. The system architecture consists of traffic signals, IoT enabled vehicles, and speed detectors. Where the

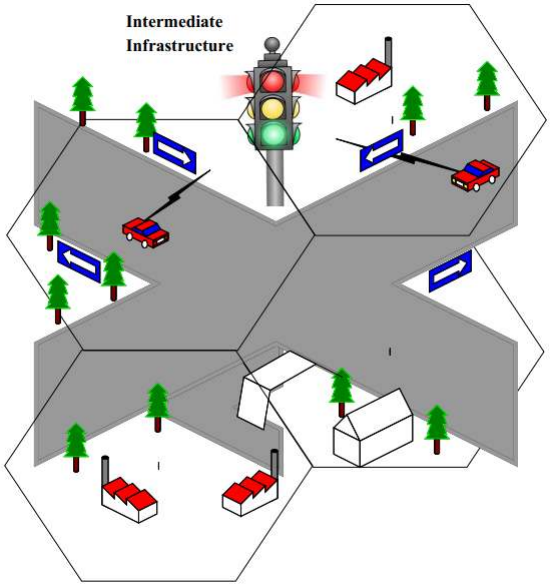

FIGURE 3. A scenario where two vehicles communicate using TBBS [31].

traffic signal acts as a BTSS and responsible for the transfer of data to vehicles. Moreover, vehicles should be IoT enabled for continuous communications and transmit data to the BTSS or other vehicles. Furthermore, the detectors are responsible to identify the speed of moving vehicles. The detectors notify the BTTS or other vehicles in case of over speeding.

Every node has a default trust value and these values are stored in a vehicle, which can be used in the future if required. Later on, the behavior-based trust value will depend on three attributes, i.e., review, sensory data, and mobile agents. The review is the data taken from vehicles. The vehicles can get information regarding the route that they expect to travel on. This information can also be based on traffic status, road accidents, expected travel time, and any other alternate available route.

The speed detector collects information of moving vehicles and creates sensory data to know if a vehicle is moving above or within the speed limit. The detection of an over-speeding vehicle helps to analyze those nodes which may cause collisions. The detector works as an event-driven function and uploads information to the BTSS or a moving node towards the BTSS.

A mobile agent starts to process its functions when a crash is reported by a vehicle. The BTSS selects a mobile agent to prove the claim. These agents may be vehicles traveling along the route, insurance agents, or nearby police cars.

This approach is a step to minimize collisions by identifying the speed and monitoring them with the help of traffic signals. The TBBS is useful for vehicles to collect data and learn from that data to establish an intelligent network. The system can be helpful if deployed in parking lots and may act as a warehouse to collect vehicles' information about a selective route. However, the TBBS is a theoretical model and at this stage it is difficult to predict its performance and effectiveness.

E. MAG-SIOT

The author introduced the concept of SIoT [32], which is based on the Alan Fiske's relationship model for the social IoT structure [33]. The relationship among social devices/objects is calculated with respect to four inter-relationships, i.e., (i) ownership object relationship, (ii) co-location object relationship, (iii) parental object relationship, and (iv) social object relationship. The main issue is how to develop trust among devices in a social relation [34]. In the proposed model, i.e., multiplicative attribute graph for social IoT (MAG-SIoT) [32], trust metrics are used to compute trust among devices. These metrics include social relationships [33], and the context in which the relationship is described [35].

The devices in SIoT possess various characteristics, such as location, operating system, and device type that create affinity among heterogeneous devices. The existing SIoT models cover only the direct relationship among objects, which cannot be created for the newly added device. To overcome this problem, the MAG model is proposed to compute trust on the basis of defined node attributes. These attributes are assigned to nodes on the basis of which the edge probability of two nodes is calculated [34].

The MAG model [34] represents node e as a vector of attribute $a(e)$. The affinity for an attribute i is a matrix Θ_i. The size of the matrix depends upon the size of the attributes taken, for example, if the value of $i = [0, 1]$ i.e., the binary value, then Θ_i is a 2x2 matrix. The probability of the link $pl(e1,e2)$ is the product of these values. In this case, 00, 01, 10, and 11 correspond to two rows and two columns. The context of a social relationship is also important and is thereby adopted from [36]. If a new object appears in the SIoT, then nodes in the co-location relationship will validate the trustworthiness of the newly added device on the basis of a triangularization method [33]. In the MAG model, unknown attributes of the newly added object are evaluated on the basis of known trusted object pairs.

Hence, the proposed model is suitable to establish the relationship based on nodes' affinity. However, the MAG model is inappropriate when the number of attributes increases as it expands the affinity matrix.

F. ATES

The Adaptive Trust Estimation Scheme (ATES) [37] is proposed for trust management of an IoT device by using both personal and non-personal trust values. When a user interacts with an IoT device, its personal trust value is calculated through the following three methods:

1) The current situation of a device is presented in the current situation vector (CSV), which is comprised of several attributes, such as device type, manufacturer, device task, and functions. On the basis of device

type, the same users' interaction history with the same type of devices is extracted, which reflects some trust level [38].
2) The obtained CSV's characteristics are sent to the server for the experience history extraction, which contains interaction history records of the same type of devices used by different users as targeted devices. The history reflects users' positive and negative interactions with the same type of devices [39].
3) The difference between situation (i) and (ii) is mapped into M5 tree regression model to obtain the trust value.

The non-personal trust value is computed through stereotypical reputation from the experience of other users [40]. However, the final trust value is dependent upon the choice of user, i.e., whether to use the concern device or not. The interaction trust value is marked as 1 in the table and -1 in case of non-interaction of users with the desired device.

The evaluation of ATES is done through questionnaires as a measuring tool. The survey's outcomes deduce the ideal trust value in case of first time interaction with a device. However, the accuracy of results depends on more number of situational characteristics.

G. TMSMD

A collection of handheld devices are connected in IoT through wireless sensor networks (WSNs), which are used for information accessing at any time. WSNs have various limitations, such as less storage, less power consumption, and limited cryptographic mechanisms. The devices connected in a WSN may face some malicious attacks, which have no proper solutions. The procedures that are used in traditional networks for a user/node's security are not suitable for WSNs [41]. Thus, it is essential to provide a proper trust and security model for WSN-based IoT. To maintain trust in the WSN-based IoT environment, it is crucial to find out malicious activities in the system. Trust management checks out faults in the network and protects nodes and network connections.

A security manager in the proposed model, namely Trust Management Model for Sensor enabled Mobile Devices (TMSMD) [41], is a single node having enough memory and computational power, as compared to other nodes in the battery-constrained sensor network. The duty of a security manager is to process authentication, integrity, confidentiality, and availability. Authentication is done via a zero knowledge protocol (ZKP) through which a user is authenticated by the security manager without revealing its secrets. The problem of accessing information and privacy occurs when a huge number of devices is connected together. The security manager handles authentication and access control mechanisms [42]. Any node can access services via access control table and is not allowed to share it with other nodes. In the proposed model, the trust is maintained at each layer of the network.

The Physical layer makes sure the integrity and privacy of data. The Application layer provides the confidentiality of services and location-aware privacy. However, because of the clear text HTTP processing, there is no specific method to overcome malicious attacks and maintain confidentiality in the IoT environment. The public key cryptography, as compared to symmetric key cryptography, uses maximum power as it is based on the integer factorization. The Elliptic Curve Cryptosystem (ECC) provides sufficient security as well as confidentiality for smaller keys in WSNs with less overhead and processing time [43], [44]. The overhead of key distribution is reduced by using the public key system in which every node has its own key to publish.

The major limitation of WSNs is that transmission relies on aggregation as there is no one-to-one link between the source and the sink. In aggregation, data is collected from a neighboring node, and after the integration of data it is sent to the nearby node for maintaining security in the proposed encryption layer. The security manager applies encryption schemes for information gathering and keeps the confidentiality of the aggregated data. A node initiates a query to the security manager, where its job is to authenticate queries and identify nodes in the network.

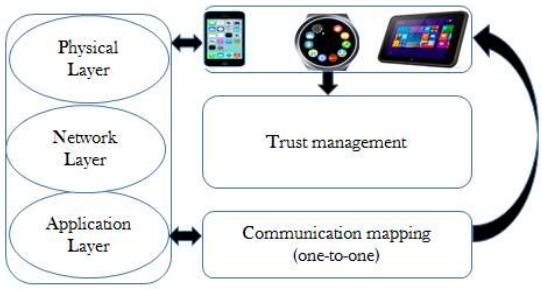

FIGURE 4. A three layered secure IoT network mode (Adapted from [41]).

In the proposed model (see Figure 4), when a node initiates a query of services from another node and there are some intermediate nodes between them, then in the first step, the encryption is accomplished through a public key. After that, the decryption is completed by using a secret key, which is finally forwarded to the node that identifies the query.

Confidentiality is provided at the time of encryption as there is no direct link between the source and the sink, thus, the proposed method introduces the data origin authentication. For the authentication, the identity-based digital signature is used to trust the security manager. The security manager produces a pair of keys and, after confirmation, sends a single key to the sink. The sink accumulates different messages and prepares a hash of these messages. The ECC receives the key by encrypting the hash and sends it to the security manager, which is ultimately sent to the source. The source then contacts the manager and decrypts the retrieved key and verifies the identity of origin. It is easy for a user to use services in the location-aware devices, however, it violates the privacy rules.

In the proposed location privacy model, an individual–having a portable device, is allowed to find the level of privacy location. The trusted security manager plays an intermediary node between users and other nodes in the network. After the authentication process, the security manager receives the query and the user then sends it to the server. The proposed model develops trust by reducing the overhead and uses a public key to protect data.

H. DTMS

A number of IoT connected devices are used to provide multiple services, where these services face serious attacks, which affect the overall communication. Malicious nodes choose selective attacks to provide a service with less processing requirement. Trust management checks out faults in the system and protects nodes as well as the network connection [45]. Establishing trust among connected devices is the main objective of the trust management scheme and it also finds the malicious behavior of a node. In IoT, several trust management schemes are proposed, such as centralized trust management [46], decentralized scheme [47], [48], and hybrid schemes [14] which depend on the application choice between both centralized and decentralized schemes [47]. The proposed trust management scheme, known as Distributed Trust Management Scheme (DTMS) [45], is based on a distributed mechanism to provide several different services in the IoT. The trust value of each node is calculated on direct observations, which is zero in the start. This start value shows that there is no trust between two nodes. This value is calculated through the discovery process by sending the announcement packet to nearby nodes. The service provided by each node has a reward if it is provided on time, and a penalty if it is not provided to the nodes. When a node sends a service request, it obtains a reward and takes a note with the allocated weight.

The DTMS performs well to evaluate selective attacks in a trust management model. The authors have calculated the trust value in a collaborative IoT network with only direct observations. Although good results can be achieved in the defense of selective attacks, however, the chances of other attacks such as Bad-Mouthing are high. Therefore, there is a need for a comprehensive trust management model, which maintains maximum trust and overcomes security-related malicious attacks.

I. SMA

In the IoT environment, as there are different types of devices that cooperate and interact with each other, swarm is a concept that elaborates the assistance of these devices to perform different tasks. The swarm system consists of different modules that provide cooperation among IoT devices in order to execute different tasks. An approach to execute the swarm process is called a semantic discovery that can either be automated discovery or manual discovery of devices. Trust is the most challenging feature in the swarm environment [15].

IoT is targeting to connect billions of devices, sensors, phones, machines, and many other products that have applications in health, operations, manufacturing, smart cities, and homes, etc. The swarm concept is applicable to such IoT applications in order to remove the underlying complexity and provide devices' cooperation to perform executions. Different components of a swarm system require to connect and maintain the trustworthiness [49].

One of the main focuses of IoT research is the trustworthiness for the relationship and cooperation of devices in the swarm process due to rapid increase in the number of these devices. There are various problems in the swarm environment, such as similar components with various names, similar components with different names, and naming conflicts that need to be fixed. Trust is important in such environments because malicious devices may damage the applications of IoT. In addition, malicious devices may also exercise trust related attacks in the IoT system. There are many proposed models in this regard, for example, [17] and [18], however, these studies do not focus on semantic discovery. In [18], when the trust is calculated, it is presumed that all devices are capable of achieving better trust values.

The Smart Middleware Architecture (SMA) [15] is proposed to provide an automatic method to identity IoT devices, calculate their semantic attributes, and estimate devices' trustworthiness. The proposed architecture is composed of two parts, i.e., the smart middleware architecture and the semantic device discovery with trust evaluation. The middleware architecture (see Figure 5) takes text attributes and data from IoT objects to calculate their semantic attributes

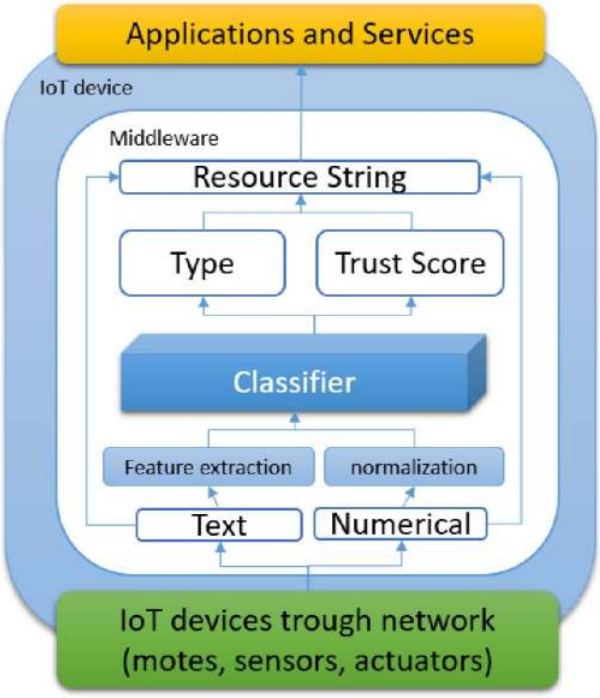

FIGURE 5. An example of smart middleware architecture [15].

and trustworthiness. After processing text attributes and data in the middleware, an output string is generated that contains the original text, classification of its attributes, and the calculated trust score of that particular device. The trust score and the class of the IoT device (calculated by middleware) help users to decide whether the IoT device is trustworthy to communicate with or not. For the trust evaluation, semantic discovery is the most important part of the SMA, which uses text and numerical information provided by the communicating device to execute the semantic discovery. The proposed middleware extracts text and numerical data from IoT devices through network. Numerical data is normalized before sending it to the classifier. The SMA uses this information for the discovery of resources and calculation of the trust score. This trust score is used for trusted communications of IoT devices.

J. DTEB

The idea of Blockchain is used for the exchange of IoT data, which is a more powerful concept as compared to the centralized data exchange. The centralized data exchange is not secure and trustworthy due to the use of a third party and the reluctance of authorities (data providers) for sharing data [50]. The three requirements of data sharing are trusted trading, trusted data access, and trusted privacy preserve [51]. The issue of trust is a challenging hindrance in the development of data industry [52]. The current data exchange platforms are unable to provide enough trust as they include third parties in the process [53].

In order to exchange data in a complete trusted and transparent environment, the concept of Blockchain, namely Data Trusted Exchanged based on Blockchain (DTEB) [51], is proposed, which was originally invented in 1991 by a group of researchers to time stamp digital documents [54]. The proposed system works on *Smart Contract* that uses the Blockchain architecture, which consists of four layers, i.e., Interactive layer, Management layer, Network layer, and Data layer, for exchanging data in a trusted environment. The basic *Smart Contract* architecture is comprised of three parts: Exchange management contract, data management contract, and user management contract. The exchange management contract further includes three kinds of contract protocols, i.e., access contract, communicational contract, and auto exchange contract. The access contract is responsible for trusted data permission management, the communicational contract records the whole data exchange process for traceability purpose, and the auto exchange contract automatically sends data access to demanders after the condition is satisfied.

Moreover, the access contract is further divided into two sections: *Data access identifier generator* and *data access right exchange*. As a data owner registers his/her data, a data access ticket (DAT) is assigned to it. The DAT enables users to access the permitted data. The *data access right exchange* is responsible for setting data provision conditions as well as implementation of automated transactions. For instance, if data providers set conditions c1 and c2, the contact will provide DAT to the demander upon the transaction request. If the transaction satisfies c1 and c2, then the demander is authorized through an access list defined for that particular data. After access is granted, users can download their desired data from particular servers. Communicational contract is determined by data demanders and data providers, and keeps record of all transactions. When a data demander sends a query, two parameters, *data name* and *data provider*, are required. The same information is also needed by data provider so that the contract is notified to both parties. When data in the *data management contract* is cached by a data provider, a separate *data object contract* is generated that records basic data descriptions (e.g., name, attribute, and data provider). Meanwhile, the *access contract* generates a data access identifier to help access the required data. For a better search efficiency, a hash table is used to perform/design customizable and extensible classifications.

The *data classification contract* includes data type management, which is responsible for the creation, modification, and storage of data objects. When users invoke the contract, they can quickly access the data set of a corresponding type. The *user management contract* controls users' security by keeping the *nickname* relationship and the *user role* relationship on the platform. The platform contains three main user roles (i.e., user provider, user demander, and auditor). The *role contract* maintains a role list that is used to define responsibilities. To avoid privacy leaks, aliases and passwords are used to interact with the system.

The proposed system is transparent and immutable to record a transaction, however, there are still privacy issues, which are the main concern while dealing with data exchange in the IoT environment.

K. ABAC

The access control technology is considered to be the most vital aspect to preserve the privacy in a variety of available networks. In the Attribute-based Access Control (ABAC) model [56], the focus is to connect nodes having data for sharing/sending based on trust characteristics. The main objective of the proposed ABAC system is to achieve the goal that the information may only be utilized for particular nodes. In addition, the data must also remain protected from malicious nodes. This model contains three modules, i.e., authentication, trust evaluation, and access decision.

The purpose of authentication is to limit illegal nodes from penetrating into the IoT. When a node passes the authentication phase, the system assigns it an authentication certificate. In the trust evaluation, the outcome of a node is obtained by calculating the trust value and comparing it with the trust threshold. After the result received, a node's trust level is managed according to the trust value. The trust evaluation algorithm is based on fuzzy sets. For the management of weight, the entropy is used, which is adjusted by experts' knowledge. The aim of using entropy is to secure the trust reliability and objectivity. The degree of a node's trust depends on trust data as well as the data that has been extracted for the trust estimation. Consequently, this

information affects the efficacy of authorization. Generally, the trust data set is represented as $G = e_1, e_2, e_3 \ldots . e_i$, and the evaluation level along with the degree of trust is represented as $G = g_1, g_2, g_3 \ldots . g_j$. After placing E and G, the fuzzy evaluation gets e_n to g_n group degree of trust. The access decision process consists of four policy points, i.e., policy execution, policy decision, policy information, and policy management. The policy execution point receives requests and executes the authorization. The policy decision point defines the access request. The policy information point stores the access control policy and executes the required actions for the policy decision point. While the policy management point is responsible for managing the access process. In addition, the policy management also has the authority to add, alter, and remove the data saved at the policy information point.

Hence it can be concluded that the proposed model provides a robust authorization as the trust level changes with the nodes' behavior. Another vital aspect of this system is a higher level of scalability and the ability of quick decision making. However, it is still not clear that how this system will perform when one node interacts with several other nodes at the same time.

L. ATBP

Adaption Trust Based Protocol (ATBP) [20] is proposed to allow security measures among nodes of a social network. The ATBP utilizes a trust policy that should be followed by all nodes in a network. This protocol is used to protect the IoT network and restrict data access. It suggests an application for travelers, known as *map guide*, which can be installed on a smartphone for trust calculation either directly or indirectly.

In the direct calculation, two communicating devices establish trust based on their behavior and mutual relationship. On the other hand, using the indirect method, the previous behavior– recorded by neighboring nodes, is used as recommendation for communications. The direct calculation is more reliable than the indirect calculation, because in the latter one the trust level makes users feel comfortable to use their private data online. It also helps users make decisions in various scenarios.

The proposed protocol considers the honesty as a trust property to cope with Bad-Mouthing attacks. It has also been proved that trust and reputation ratings of individual nodes in distributed environments of social media are effective approaches for improving security, decision making support, and promotion of collaboration among nodes. Moreover, all nodes maintain friend lists thanks to their trustworthiness. Therefore, when two nodes communicate with each other, they exchange their friend lists. This strategy supports android travel map guide to decide the best route according to current situations, road conditions, and recommendations.

The ATBP is useful in deciding the best route so as to avoid traffic congestion and accidents, and provide safe and smooth drive. However, it is not confirmed that how trust can be calculated from the mentioned properties and how accurate it will be. Furthermore, the travel map guide application gives an easy access to applications, but it may be hindered in high dynamic situations.

M. DCTEPF

In the IoT network, subscribers always interact with a huge amount of information wherein they are unaware of the source of information that they receive. In such situations, trust plays an important role in providing reliable services. To build the trust of an entity, metrics and attributes are defined on the basis of reputation, experience, and knowledge (REK) [57]. These attributes are then combined and some methods are utilized to assess recommendations and trust [58]. Knowledge consists of data obtained from the first party and is computed by various attributes such as temporal attributes, i.e., frequency and duration of interactions, relationship attributes, e.g., cooperativeness, co-location, parental, and co-work. For the intelligent decision making, trust is computed based on two other metrics, i.e., social and nonsocial trust. The prior is calculated based on persistence, willingness, and confidence [59], whereas the later is defined on the basis of disposition, competence, dependence, and fulfillment [60].

The data centric trust evaluation and prediction framework (DCTEPF) [55] consists of various modules, for example, trust data access object, trust service enabler, decision making and prediction, data repository, TrustComputation, trust agent, and Application Programming Interface (API), as depicted in Figure 6 [55]. Trust metrics related to both data and entity are calculated separately in a module called trust metrics extraction. Data requirements are identified by decision making and trust computation prediction modules. Trust computation happens parallel with various kinds of models such as machine learning model, numerical model, and prediction model. Trust agents are responsible for information gathering and computing. Then, significant data is gathered and computed in the data access object (DAO) module, and is kept in a warehouse for further utilization. Moreover, important attributes are estimated and combined on the basis of REK trust model, while decision making takes place based on these attributes.

Data trust is calculated by considering some attributes, such as the ratio of records, the cost of task execution, the success of task execution, and the time difference between information accuracy and. An algorithm is used for the trust prediction of various subscribers and trustees, where attributes are provided to the algorithm for the trust computation. This system is useful in computing trust between information sources and a trustee whether there is no prior interaction between them. The system is also useful to filter incorrect data. However, it is not suitable for handling contextual data for the prediction of trust.

N. TrustCEP

The Internet technology is shifting gradually and physical things are getting fastened to ramp up ubiquitous networks,

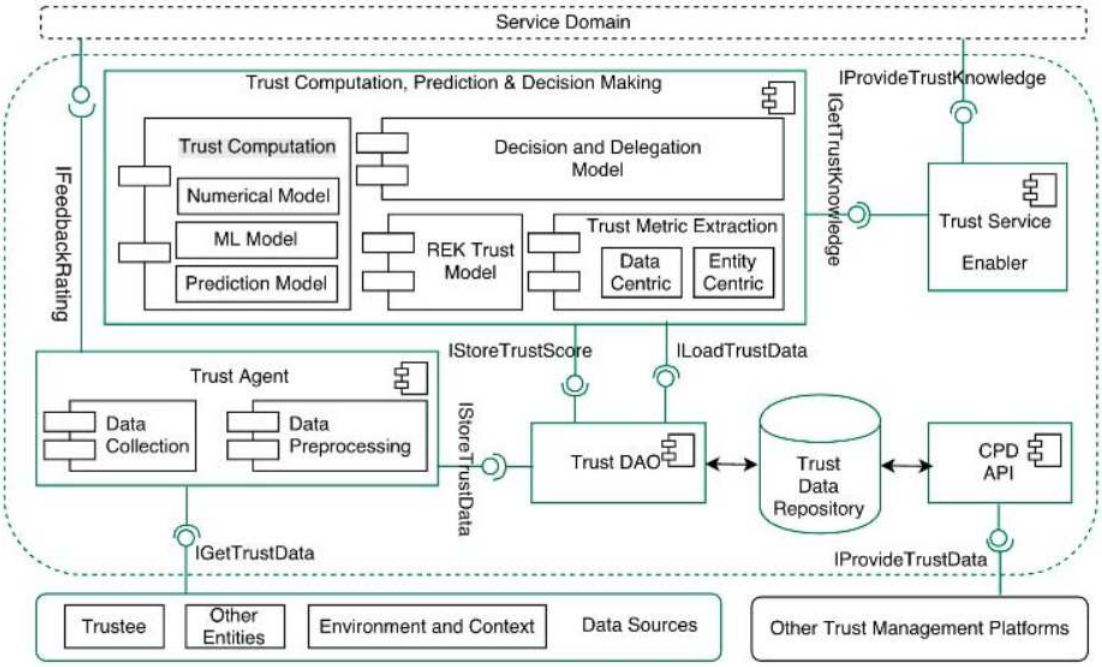

FIGURE 6. Data centric trust evaluation and prediction framework [55].

i.e., IoT. In IoT, various sensor-equipped devices are connected together in order to collect as much information about their surroundings as possible. It provides a platform for context-aware computation by allowing interaction among different users and applications. There is a need for the identification of events, determining their pattern, and analyzing their meaningfulness/impact, which are achieved through the Complex Event Processing (CEP) [61].

The CEP is used to extract complex contexts in different event streams by analyzing their patterns. While studying CEP, some terminologies are used frequently, which need to be explained.

Operator: It is a computing module.

Function: It is used for the analysis of input event streams such as aggregation and filtering [62].

Operator graph: It is a tree-like structure, called directed acyclic graph (DCG), and is responsible for the connection between the producer and consumers. In event streaming, this graph decides that which operator should execute first [63].

CEP can be placed on one of many devices and thereby forms a distributed system that shows device-to-device communications [64], [65]. One of the major drawbacks in the CEP operator placement is a privacy threat. When contexts get processed in a distributed environment then privacy issue becomes crucial. Thus, in order to deal with the privacy of data, a privacy-aware mechanism is needed. The proposed model is influenced by the combined concept of three existing models, i.e., CEP model, system model, and adversary model.

According to the *TrustCEP Model* [61], users are categorized into two classes, i.e., the producer and the consumer. The producer and consumers are connected through DAG, generally termed as operator graph, which includes the collection of operators. These operators serve as computing modules and prescribe the events. This model was proposed to show some sensitivity level when an event occurs. This sensitivity level helps to determine the desired privacy of users. This model aided the assumptions that every user has to define their own sensitivity level according to their privacy constraints and device specifications.

The second source, from which the idea was leveraged to support the proposed architecture, is *System Model*. It helps to assume that all devices in a network must be viewable by other devices during the execution of a specific query. No single hidden entity is allowed to become part of a network, which can violate the privacy of other devices. Moreover, the movement of all devices is assumed to be not highly dynamic so that they will remain in a viewable range during a specific operation [63], [66].

The third model to support the proposed work is *Adversary Model*. From adversary model, the idea of operator placement was taken only on glorified devices [67]. The main focus is on two primary attacks, i.e., *collision attacks* and *On-Off attacks*. Collision attacks are those attacks in which some devices collectively try to control the system and hack private information. In On-Off attacks, a device swiftly gets turned into despiteful entity while playing a malicious role.

Trust based distributed CEP is based on mutual trust between two users. The proposed approach involves the following steps:

- Measurement of mutual trust
- Improvement of mutual trust between two users
- Algorithm for analyzing privacy awareness

1) MEASUREMENT OF MUTUAL TRUST

The strength of the relationship between two users is based on their trust, which can be measured by the data shared between them [68]. Data shared can be analyzed by looking into their interactions either from synchronous (calls) or asynchronous (messages) channels [69], [70].

2) IMPROVEMENT OF MUTUAL TRUST BETWEEN USERS

In this phase, the trust vector is shared in users who analyze the trust vector and give a recommendation. Then the following approach is followed:

- Determine the credibility of recommendation
- Detect changes in behavior
- Update existing values with recommended ones

This assessment is based on similarity measures of distance between trust vectors. Based on trust values and recommendations, users can assess the divergence in trust values. If both have the same divergence of trust values then the data can be adjusted. For this purpose, conservative increase and multiplicative decrease principles are used depending on the valences of divergence. As a result, every user looks into the divergence of his recommendations as compared to other users' recommendations. Moreover, every user should penalize other users who are giving malicious trust values and as a result, the trust value of that user goes down. Similarly, if a user is not giving any falsified values then his trust value increases for the loyalty.

In On-Off attacks, those users who give falsified values are marked as benevolent users, and after a certain threshold limit they are marked as *malicious users* and are then penalized.

3) TRUST BASED OPERATOR PLACEMENT

With the help of algorithm, a trust-based CEP graph is obtained. Initially, every user tries to find neighboring users, looks into their trust vectors, and based on privacy constraints they adjust their recommendation vectors. If there are no neighboring users then the graph is initiated on their own device and requests are placed for *collaborating placement requests*. Upon a conflict in the request, a conflict message is initiated if there is any user who can look into the path conflicting and give the recommendation vector. The initiator looks into the request, modifies the graph, and finally executes it.

Besides, the proposed model fails to provide support in a scenario where a high mobility is involved. It is assumed that dynamic devices will remain in the range of execution, however, in a real scenario, devices may be highly dynamic. Therefore, additional mechanisms are required to deal with a high mobility of devices in a network [63], [64].

O. MAPE-K

The integration of IoT with Cloud computing provides convenience in the complexity and reduction of cost as Cloud computing has different capabilities such as processing and storage, and emerges as a mature technology [71]. Because of the autonomic and dynamic nature of the Cloud, environmental trust management is a difficult task. The deployment of Cloud of things is complex because of low computational capacity [72].

Existing studies focus on trust management without considering the autonomic and dynamic nature of the IoT environment. Autonomic computing means to equip devices to show an adaptive behavior towards the dynamically changing situations such as self-configuration, management, protection, and openness. Trust is also considered as a service [73], however, the calculations of feedback management is a difficult task due to the unpredictable number of devices and their autonomic behavior. A MAPE-K (Monitor, Analyze, Plan, Execute, Knowledge) loop based autonomic trust management framework [72] is proposed to handle dynamic environments. It is helpful for the adaptive trust management in the IoT Cloud environment and provides facility to tackle malicious recommendations from other devices.

The concept of distributed trust agents is used in MAPE-K feedback loop to provide quick response. The information is aggregated and filtered in the monitoring module. In the Analyze module, all gathered information is evaluated by agents, and if any changes are required then it is passed to the Plan module. The Plan module performs actions on the collected data to achieve the desired result. Actions recommended by the Plan function is managed in the Execution phase based on device behaviors. The Trust Executor acts as an open API in the MAPE-K framework. The processed data from all phases is known as knowledge, which is used for decision making and is shared among distributed trust agents. The data includes all contextual and topological information. System now handles dynamic issues and becomes self-adaptive. The proposed model gathers environmental and contextual information from the IoT environment and sends it to the MAPE-K loop framework.

The proposed architecture is comprised of three layers, i.e., Cloud network layer, Service consumer layer, and Applications and service layer. In the Cloud network layer, intelligent computing is used to obtain related parameters. These parameters are used for decision making from the MAPE-K loop. The Service layer consists of APIs that facilitate client access towards the services and filter the required information.

Below are the key features of the MAPE-K framework:

- *Availability*: It shows the reachability between the Cloud system and the target environment.
- *Scalability*: It refers to the capability of handling IoT devices.
- *Accessibility*: It denotes the ubiquity of services through the Internet.
- *Flexibility*: It signifies the distributed trust agents that provide flexible environment to handle dynamic situations.

The proposed model contributes in increasing the dynamic trust management level by using a self-adaptation method.

However, the problem occurs when the service disrupts or the data attributes increase from its limit. The model may also have some problems in handling malicious nodes.

P. TDFDS

With the advancement of smart devices and popularity of IoT, the number of connected devices to the Internet increases on a daily basis. The huge amount of data that is produced by these devices would be stored on clouds [10] and used by different distributed systems. By storing data on clouds, there are certain security threats that need to be resolved [74]. It requires a proper framework to develop trust between humans and machines. For this purpose, a lot of effort has been done by researchers to create trust among distributed systems and to compensate challenges of interoperability [75]–[78].

For complex interactions, two main approaches are used for trustworthiness, i.e., policy-based and reputation-based. Policy-based approaches are time independent that are mostly used in firewalls to authenticate the user access. On the other hand, reputation-based models use gray-levels based on the history of experiences faced by other users. Another way to represent trust is to use PROTUNE predicates by using trust function having parameters of trustors and trustees [79]. However, a framework is needed that can evaluate trust in dynamic and different situations.

The proposed Trust-based Development Framework for Distributed System (TDFDS) [74] has four main pillars, which define different variables of trust and each pillar is put on PROTUNE to test trust. These pillars include environment, customer, business requirement, and technology.

- *Environment*: It contains technological attributes as well as social, cultural, and religious factors.
- *Customer*: It consists of system and human intelligence as well as habits, genders, and physical abilities.
- *Business Requirement*: It includes attributes that effect the trust.
- *Technology*: It comprises security and usability.

It is to be noted that the trust evaluation framework is integrated with applications as a separate trust evaluation module for the cost effectiveness.

Since every web application runs on the server, the server related hardware should be trustworthy. The hardware can be made trustable by working on physical attacks, malicious node attacks, reverse engineering, and hardware processing speed [80], [81]. Similarity, in the software platform, application server, operating system, and database management system cooperate with each other for the enhancement of security and usability [82] to meet the software trust level.

The proposed trust framework provides trust for online integrated and distributed applications. It also addresses security threats for applications, which have different nature of problems. The security trust model defines trust levels through variables that work for different distributed systems. Besides, the model addresses only limited risk vulnerabilities and does not take into consideration the security related to unknown risks of the Cloud. During the running of an application, it does not offer security mechanisms for administrators.

Q. CBSTM-IoT

The Context-based Social Trust Model for the Internet of Things (CBSTM-IoT) [21] is designed to increase the collaboration among trusted nodes and limit the interaction of suspicious devices. In the CBSTM-IoT model, node transaction factors and node social relationship factors are two basic components for trust calculations. The node transaction factor relies on the following four parameters:

- *Context Importance*: It refers to how often an interaction in a specific context may occur. More context specific interaction leads to more weight.
- *Node Computation Power [83]*: It signifies the decrease in computational power and capability of a node to act maliciously.
- *Confidence [84]*: It reflects that how many recommendations of trusts are coming from other nodes to a particular device.
- *Feedback*: It shows the performance evaluation of a node after the transaction is completed. Each node has its own evaluation for every transaction.

Similarly, node's social relationship is the combination of the following two parameters:

- *Owner Trust*: It depends on two factors, i.e., a) friendship, which refers to the entire mutual friends of two owners, and b) centrality that depicts the node's influence in a social network.
- *SIoT Relationship [28]*: It expresses the type of relationship among nodes characterized by relationship factor.

A single owner having two nodes refers to a high relationship value, while no relationship among nodes depicts a low relationship value. In the CBSTM-IoT, the higher relationship value indicates a higher trust. The CBSTM-IoT integrates all trust factors for the trust calculation at the time of nodes' interaction. Trust is calculated for each node by a specific value in the range of [0-1], when two nodes have interaction with each other. These calculations are based on *Direct* and *Indirect* trust.

Direct trust is calculated when the direct interaction takes place between two nodes. That is, if node i and j have no interaction earlier then it means that there is no trust value as there is no relationship between them. Thus, the relationship value is set to 0.5. The relationship value is set to 1 if node i and j have already interacted in any context. Conversely, the Indirect trust shows the recommendations of other nodes for node i to interact with node j, if and only if node i has no interaction with j in any context. Node i calculates the trust value by the recommendations of other nodes having interactions with node j in a specific context.

The CBSTM-IoT is an adaptive model that can adjust itself according to the behavioral pattern changes in the IoT environment. In addition, this model does not depend on specific nodes and peers. However, there might be some malicious

nodes that may behave fairly sometimes and allocate higher trust values to other nodes as indirect recommendations. Thus, the more indirect recommendations, the more decrease in the accuracy and performance of trust calculations.

R. TIMELY TRUST

The Timely Trust framework [85] identifies the demand of IoT in global virtual teams (GVTs) and tells how the swift trust formation in GVTs is affected by different cultures. As the world is becoming ''Global Village'', multinational corporations must rely on the IoT environment to increase the performance of GVTs. A GVT is a team that consists of people from different countries and geographical regions having differences in their cultures, languages, and time zones, which are grouped together to perform a specific task. GVTs have less face interaction but highly dependent on communications technology [86]. GVTs have common shared objectives on which they work across geographical boundaries and depends on technology such as computers to communicate. They do not have any previous working record with each other and also have cultural differences.

The IoT has given soul to different electronic devices to make human life comfortable with respect to many applications, for example, smart cities, smart offices, smart healthcare, smart transportation, and smart communication systems [19], [87], [88]. Computing devices and the Internet, known as IoT, provide a way of electronic communications between people and devices [10], [89] and therefore make the work structure of GVTs more effective. The main objective is to trust strangers that collaborate in GVTs to work in a distributed global environment. Trust should be developed rapidly so that teams around the globe perform their functions effectively. One of the goals of GVTs is to complete tasks and projects fast and effectively within a prescribed time.

As mentioned earlier that GVTs are formed from different cultures, trust behaviors must be considered that depend on cultural values. Some cultures first establish relationship among team members to perform efficient tasks, while others focus on completing their projects regardless of relationships [90]. The idea describes the IoT usage with respect to cultural differences and motivates the use of IoT in a virtual global structure.

IoT paves the way for sending and receiving information at a geographical distance where connectivity exists. Thus, the concept of smart devices provides fast and successful communication among all GVT members [91]. The GVT members use cloud servers for data storage and create efficient project management [91]. Due to these embedded IoT concepts, the GVT members can easily communicate by using video calls or voice messages and also access data from shared storages.

Besides, privacy and security are major challenges in the interconnectedness of thousands of IoT devices. People will no more trust when they come to know that their record and data may not be safe on the Internet [19]. In addition, when data is stored on remote servers and is shared over different regions, there may be chances of cyber-attacks, as happened recently with Facebook [92], and therefore the GVT members may lose trust on smart services.

S. DTRM

In the IoT network, malicious nodes open up the issues of users' security and privacy [93]. In order to avoid these malicious attacks, several cryptographic algorithms have been designed to provide users' security and authenticity [46]. The IoT devices may differ in computational power, storage capability, communication ability, and sensitivity. A distributed trust and reputation model (DTRM) [94] is proposed, which is the extension of [46] and [95], to overcome their limitations and classify trustworthy communication channels. Instead of using a centralized environment for trust calculation, the DTRM focuses on distributed environment to make IoT devices capable of handling processing by themselves. The model also proposes different levels of security, which are suitable for sensitive devices in the IoT environment.

This model classifies stronger devices as ''Alpha'' nodes, which are declared and configured during the model setup. The alpha nodes in the system are responsible for assigning jobs, collecting data, and profiling of IoT devices. In addition, it keeps record of all devices and manages them according to their requirements. Some devices share similar data where their functionalities are grouped together in the form of virtual clusters. The sensitivity of data is very important factor in the IoT, therefore, the proposed model defines different levels of security, which minimizes computing power for transmitting publically available information. Some information needs public access, for example, weather broadcast, while others are kept confidential such as patient record. Thus, different levels of security are defined in the proposed model. To handle this scenario, the DTRM uses flag parameters. If flag is 0 then information is publicly provided, while it is 1 in the case of confidential information.

The proposed model is classified into seven different phases to perform smooth computations in the IoT environment. The first phase is data collection where a detailed collection of information is stored by alpha nodes. In the second phase, virtual clustering, similar devices are identified based on collected information and are placed in a virtual cluster. In the third phase, weight calculation, virtual clustering is exercised to find device recommendations so that to calculate the trust values of particular devices. The quality of recommendations helps the system to give trust score to IoT nodes. The fourth phase is transaction, wherein the flag is set to either 0 or 1 on the basis of data collection in the first phase. The fifth phase is called trust computation where trust scores are calculated for each device. The sixth phase is node classification in which a proper decision is taken regarding devices, i.e., to know if a particular node is trustworthy or not. This decision is based on summation and averaging of the output of all neurons in a network of similar class [96]. The last phase, rating update, compels nodes to rate other devices on the successful completion of transaction. The rating ranges

from 0 to 5, where 0 indicates a bad experience and 5 shows the best experience.

The proposed model provides protection against Bad-Mouthing, Good-Mouthing, and ballot attacks as it mainly focuses on classifying trustworthy devices from malicious ones. However, some attacks are difficult to handle, such as distributed denial of service (DDOS), Man-In-The-Middle (MIM), and wormwhole attacks.

T. ANTs

The proposed concept of Application-Driven Network Trust Zones (ANTs) [97] classifies the network into trust zones that depend upon the application layer, where communications take place in the medium access control (MAC) layer. By doing so, there are certain benefits, for example, a trusted device can be used to check new nodes and reconfigure the existing trust zones. The formulation of trusted zones restricts remote communications, thus, the damage caused by malicious nodes can be limited and the overall network is saved from several kinds of attacks. With a huge number of devices in a smart building, the probability that a device is compromised cannot be calculated easily, but it can be assumed that increase in the number of devices upsurges the probability of attacks.

The proposed model puts all devices in a separate network at the MAC layer so as to reduce the impact of malicious nodes. The trusted zones may contain only those devices that are present at the application layer. This phenomenon prevents sensitive nodes from malicious attacks in the network. The single trusted device in a network can monitor the descriptions of all participating nodes in a smart building [98]. In addition, this device can also act as a certificate authority to authenticate all nodes in a smart building.

The ANTs scheme can reduce the unwanted and malicious communication in smart buildings. In case of finding any compromised node in a network, the trusted device creates a new set of credentials to minimize the attack's ratio. The general concept of ANTs involves IEEE 802.11s standard [99], WPA/WPA2 [100] and SAE [101] protocols. The application layer uses machine-to-machine (M2M) protocol, named Constrained Application Protocol (CoAP) [102]. The embedded devices used in smart buildings are termed as end devices (ED) in the proposed concept of ANTs. The trust zones can form independent network and EDs in the network can be configured frequently by using SAE, IP, DTLS, or CoAP protocols. The ED becomes part of the network and configures more joining nodes to a smart building network. These devices then communicate over a secure channel using Smart Home Gateways (SHGW). The SHGW classifies nodes into trusted zones by checking similar attributes. It may also work as a monitoring device to check the behavior of all EDs, identify malicious nodes, and exclude them from the network. However, designing proper strategies and routines to put EDs into the trusted zones is a challenging issue.

To overcome this problem, an algorithm must be designed for putting similar EDs into the trusted zone, excluding malicious nodes immediately, and reconfiguring all devices in the network. The suitability of ANTs has not been tested in a real network scenario.

U. TAS-IoT

As IoT has a low powered dynamic environment, it is difficult to provide efficient security services by using static security mechanisms because it consumes a lot of resources whether required or not. Different mechanisms adequate for different situations focus on adaptive security [103]–[107], for instance, an adaptive game based security model [104]. Trust management embodies the concept of confidence between two nodes, i.e., trust between a trustee and a trustor [14].

In the proposed Trust-based Adaptive Security in the IoT (TAS-IoT) model [108], nodes are categorized into two classes, i.e., legitimate nodes, which are allowed to post messages, and non-legitimate nodes that are not allowed to post messages but some time they post bogus messages to disturb the communication. To authenticate messages, an authenticator is appended with every message by the legitimate node. It prevents non-legitimate nodes to post false messages in the network and reduces power consumption by authenticating data at its origin. On the basis of trust between two nodes, each node decides whether to authenticate the received message or not. A trust value is associated with each node that has values between 0 and 1. This trust value is associated on the basis of observations, experience, and recommendations. After the trust value is calculated, an adaptive function is used to decide whether to authenticate the message or not. If the trust value exceeds the threshold value then it means that the node (trustee) wins the trust of another node. Thus, the receiver (trustor) decides to legitimize messages from the trustee with no further need to specifically authenticate them.

On the other hand, if the trust value is less than the threshold value then the receiver decides that there is a need to manually authenticate the message from the sender rather than legitimizing it.

V. CTM-IoT

Centralized Trust Management Mechanism for the Internet of Things (CTM-IoT) [109] is proposed to offer reliable information sharing among IoT devices. The model comprises a super node to function as a centralized trust manager (see Figure 7). The IoT environment is divided into clusters for achieving trustworthy communications among different devices, where each cluster includes a trust manager, known as master node. The super node stores trust data of all master nodes and cluster nodes in the central repository. The super node is also responsible to monitor various activities, such as traffic management of the entire network and trust management among all IoT devices. It also sends and monitors Internet data packets between the master node and cluster nodes, and the IoT applications and the master node. An IoT application can request information regarding a particular cluster node by simply sending a request message to the master node

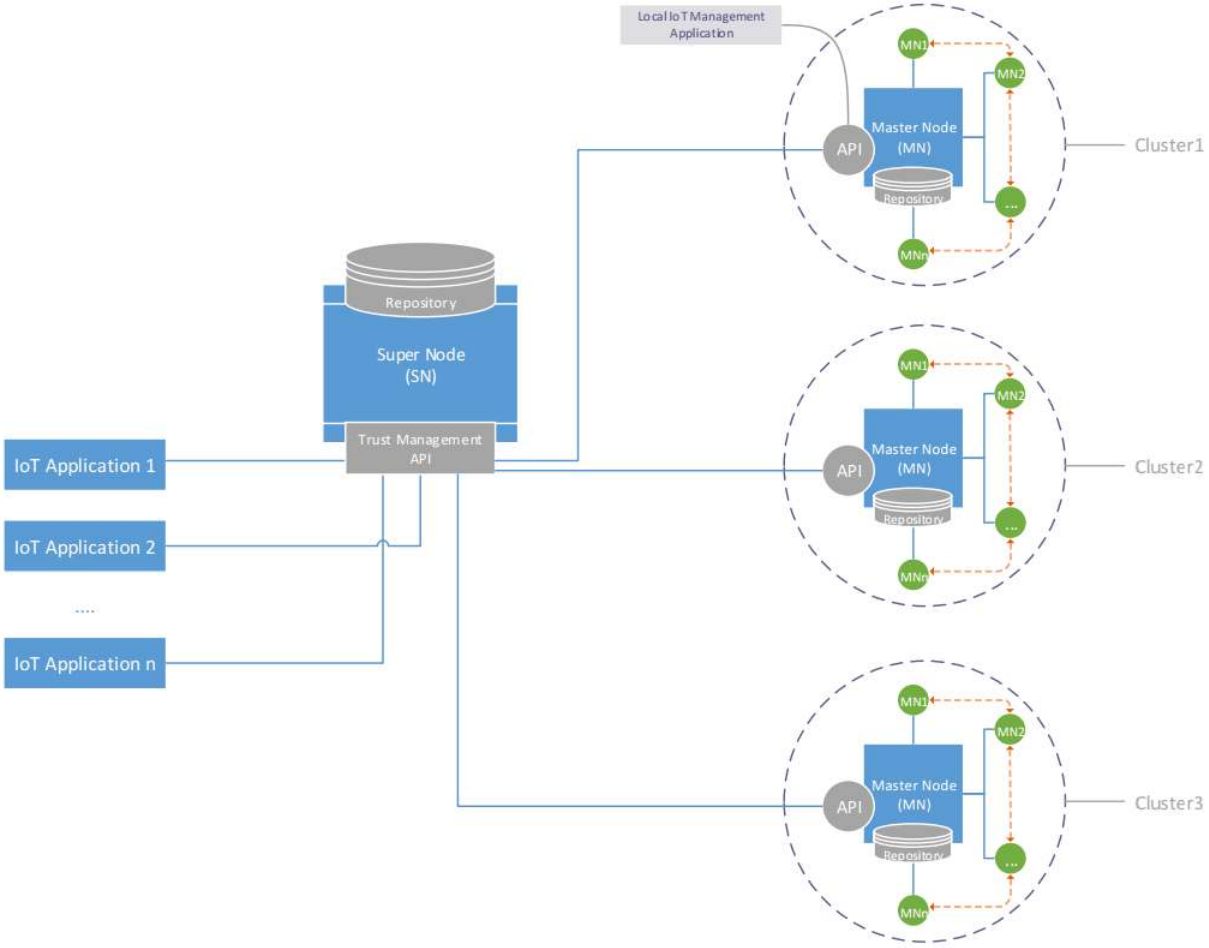

FIGURE 7. CTM-IoT that consists of three clusters, where each cluster includes master nodes; three IoT applications; and a super node that stores trusted data of all master nodes [109].

of a specific cluster. It can also provide information if any cluster node or IoT application requests certain data.

Furthermore, the super node has a repository wherein trust values of all master nodes are stored along with their addresses. The repository functions as a routing table to record the trusted information as well as structure of the network, and supervises all devices that which one has to join which cluster in the CTM-IoT framework. The primary goal of the CTM-IoT is to provide trust among IoT nodes that can be achieved through super nodes, which are the main trust manager of the entire framework. However, without comparison with existing schemes, it is early to predict that the proposed approach outperforms others.

W. TMCoI-SIOT

A trust management system based on communities of interest for the Social Internet of Things (TMCoI-SIOT) [110] has been proposed to incorporate various features such as social modeling of trust, direct and indirect trust, and transaction factors. The proposed mechanism focuses on the social Internet and integrates several parameters of trust on the basis of direct and indirect evaluation. The TMCoI-SIOT architecture utilizes the concept of clustering and divides nodes into communities based on interest [111]. The architecture of the TMCoI-SIOT is shown in Figure 8, where a network has a dedicated SIOT server, nodes that are clustered together as a community, and the trust administrator to manage the security of SIOT.

The formation of a community begins by the authentication of a node. If a node wants to join the SIOT then the SIOT server authenticates it. After the authentication, the node is allowed to join the community of its own interest or either it can start creating its own community. Furthermore, every community has their own unique trust administrator. The selection of a trust administrator is based on trust and the parameters used for trust evaluation are comprised of trust level, capability, and sociability of a node.

After the selection of trust administrator, the re-selection of administrator begins if admin becomes malicious, loses its connection with nodes, leaves the entire community or an authorized area of the community. The responsibilities of admin include the calculation and storage of trust values. When a new node sends a joining request to the admin then it calculates the trust and compares it with the threshold

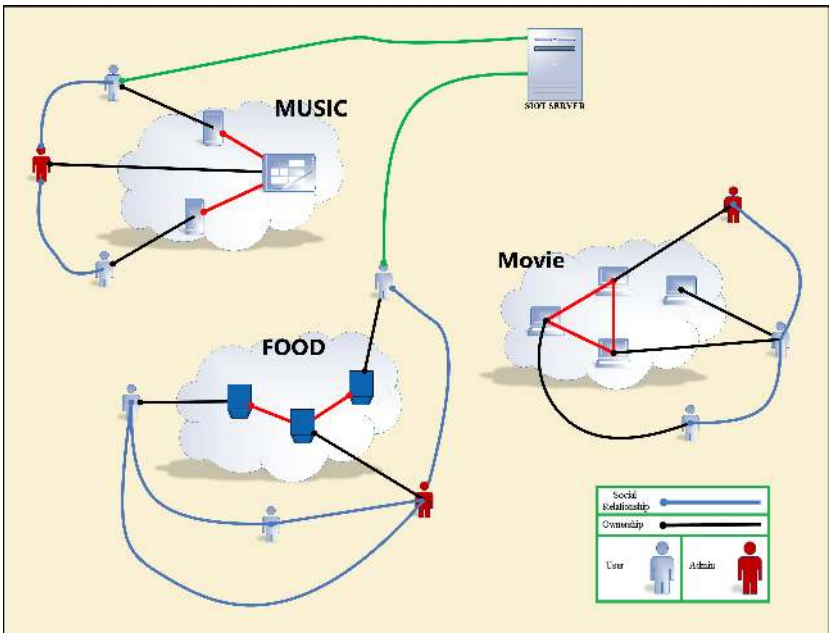

FIGURE 8. The TMCoI-SIOT architecture [110].

value. If a trust value is greater, the admin examines the node's similarities and geographical area to accept the joining request.

The strength of TMCoI-SIOT is such that its architecture is based on clusters, which helps to reduce challenges associated with the memory storage. The proposed scheme is evaluated against On-Off attacks, however, the effectiveness of the proposed mechanism is uncertain for Good-Mouthing as well as Bad-Mouthing attacks.

X. CTMS-SIOT

SIOT is a flourishing area of research that arose due to the merging of social networking concept and IoT, and caused the birth of advanced and sophisticated applications [112]. Trust management system (TMS) being the most reliable security mechanism puts forward the need for research regarding trust prediction and evaluation [113]. The TMS considers historical behavior and entity rather than its context. Nodes' heterogeneity and mobility make IoT incompatible with the available Internet solutions.

The Context-based Trust Management System for the Social Internet of Things (CTMS-SIOT) [114] was proposed to consider dynamic trust values along with relative context in different tasks to keep a realistic approach. In this model, the nature of architecture (centralized or decentralized) is based on computational complexity, where a node's life time decreases due to data storage in a decentralized architecture. The model contains variable objects, user compatibility based service, and a trust management system considering a contextual and feedback approach. It consists of two modules, which are responsible for contextual trust and reputation in addition to behavioral categorization and prediction. Furthermore, the CTMS-SIOT also includes a feedback system to evaluate certain or uncertain transactions, transaction weight, computation capability weight, and context weight.

In the trust computation process, initial trust values are assigned by neighbors based on their mutual relationship. Object relationships can be of three types, i.e., ownership relationship, domestic relationship, and social relationship. In the owner relationship, the objects are introduced in the SIoT by a particular user. The domestic relationship involves objects that have a common workplace. While the social relationship is formed due to mutual interest. In the first two relationships, no malicious nodes can be found, which result in higher trust values.

The trust request from a user triggers the discovery mechanism, where lack of history in the local trust table compels a user to send a trust request query to the server. Upon the user request, the trust manager initiates the entity selection process on the basis of past interaction or prediction of compatibility using a decision tree algorithm, for example, Quinlan C4.5 [115]. The proposed model is used to calculate social similarities between the requester and the selected node by comparing similar and different aspects of the sample set, which include friendship similarity, community interest list similarity, and object profile similarity. After calculating similarities, the system initiates the assessment of nodes' credibility. Finally, the evaluation of transaction occurs, which considers requester's satisfaction feedback and server level that measures contextual trust and reputation. The CTMS-SIOT carefully controls a dynamic environment while providing efficient services. However, it depends on the past interaction or prediction of compatibility, which may be wrong and therefore reduces the system trustworthiness.

Y. TMF-VSN

In Vehicular Social Networks (VSNs), vehicles, infrastructure, and connectivity points, i.e., road side units (RSUs), are the core components. To guarantee a trusted communication, these modules develop an essential issue. A trust-based model, known as trust management framework (TMF), is proposed for VSN– know as TMF-VSN [116], which is based on three layers of trust for the VSN environment. These layers are interconnected in three levels and work mutually. Global Trust Manager (GTM) lies on the top level that holds the authentication of vehicles' profiles in a network. Also, it maintains a profile list about the data of individual vehicles, which include two lists, i.e., history trust list and friend list.

A new subordinate of the system is Domain Trust Manager (DTM) that holds the history, domain, and relationship profiles of each individual vehicle. The entire communication takes place between Vehicle to RSU and RSU-to-RSU. Each vehicle maintains its information in its own list, namely Vehicular Trust Monitor (VTM). Each vehicle must maintain four types of information regarding the lists, i.e., internal friend list, direct neighbor list, indirect neighbor list, and history trust list. Moreover, the TMF-VSN comprises four modules, i.e., friend trust, neighbor trust, global trust, and history trust modules. In the friend trust module, the trust evaluation takes place into two parts, i.e., external friend trust and internal friend trust. And thus the vehicles are also divided into two lists, known as external friend list and internal friend list.

In the neighbor trust module, the trust evaluation occurs where neighbors are divided into two lists, i.e., direct and indirect lists. And therefore the trust should also be either direct or indirect based on the neighbor nodes' placement. The history trust module provides a global trust representation of vehicles, where the calculation of friend trust, neighbor trust, and history trust is combined in the global trust module.

The proposed system can improve the network performance by increasing the packet delivery ratio in the RSU and therefore decreases the end-to-end delay. However, the validity of experiments may be affected by some factors, such as the number of packets by a requesting node and the density of nodes.

Z. IoT-HiTrust

For establishing trust in the hierarchical IoT management, three level cloud hierarchical mobile model is presented in [118]. In this model, the topmost layer includes servers, the middle layer consists of cloudlets (heavyweight devices), and the bottom layer comprises lightweight devices. The cloudlet devices have a reasonable computation as well as storage capacity to unburden the bottom layer devices. The cloudlet devices only move inside a cloudlet region and are linked with the Internet. The bottom layer devices, due to mobility irregularly, are linked to the Internet with a carrier and move from one cloud to the other. A disconnected device creates a connection with its regional cloudlet for taking services without any break with wireless communications. The communication between IoT devices and cloud is within the regional cloudlet owing to physical juxtaposition that saves power and bandwidth. The cloud serves as a logical entity containing a number of cloud servers expected security and protection. In the Cloud Hierarchical Trust Management for IoT, namely IoT-HiTrust [117], the trustworthiness of all IoT devices is calculated by a cloud in the region of cloudlets. The cloud grants permission to resources and brings heavyweight IoT devices near the cloud. The selected heavyweight cloud devices that control the cloudlet region are known as cloudlet devices.

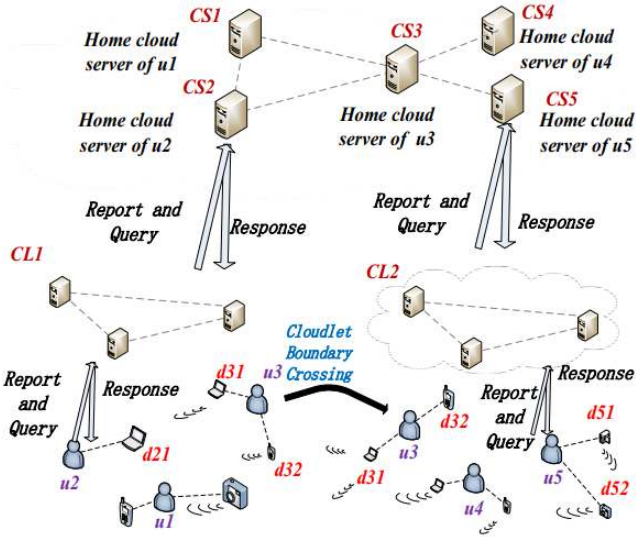

FIGURE 9. The IoT-HiTrust architecture [117].

In Figure 9 [117], two cloudlets, i.e., CL1 and CL2, having three IoT devices, work as cloudlet devices. The top layer's node simply refers to as cloud, the bottom layer nodes are the IoT devices, and a node in the middle layer is called a cloudlet. Every user/IoT device at the cloud service level is recognized through a unique ID. For load balancing, a user with a unique ID is allocated to a cloud server, known as *home*, which is used to manage users' data. The home cloud server of a user remains the same, however, its virtual machine (VM) may transfer from one place to the other. If the user data is asked from users' VM, then the request is fulfilled by their home cloud. In case, each owner has multiple devices, then all devices are mapped to the owner home cloud.

The requests and replies of IoT devices are transmitted only inside their own cloudlet region along with their cached trust information. If the Internet connection is terminated then a cloudlet replies user queries within the region with a disconnection mode [118]. Moreover, if a user moves from one cloudlet to the other then it is removed from the previous cloudlet and is registered in the new one. Also, in the previous cloudlet a pointer is added so that user's reply can be sent to the new cloudlet.

The trust model is centered on social relationships among IoT device owners. A receiver calculates the trust value upon recommendations of the recommender by applying recommendation rules, which are created on the basis of social relationships. For measuring social relationship, the following three main social metrics are used:

- Friendship (demonstrating intimacy): If two owners are friends, then they cooperate with each other. The IoT-HiTrust does not distinguish friends from acquaintances.
- Social contact (demonstrating closeness): Devices' owners with same mobility patterns have high chances of social contact.
- Community of interest /COI (demonstrating subject matter knowledge and standard): Nodes of same groups likely to share same interests [112].

Users of the same social relationship have identical views regarding the services delivered by an IoT device. The transformation of the social relationship between two device owners is based on the social similarity of those devices. A malicious device owned by a malicious owner can cause a number of attacks. The IoT-HiTrust controls self-promoting attacks, Bad-Mouthing attacks, ballot stuffing attacks, discriminatory attacks, and opportunistic service attacks, which disrupt the trust of a system.

The IoT-HiTrust approach achieves appropriate trust properties by tackling attacks resiliently in a large IoT system. However, the system fails to control intruders as it does not take into account the intrusion detection. For more understanding about intrusion detection, interested readers are referred to [119]

A summary of charachteristics of the reported techniques is presented in Table 1.

III. LESSONS LEARNED

In the IoT system, devices perform various tasks to handle real life situations without human intervention. Therefore, without human interactions, it is indispensable to have a strong and reliable communication among these devices. In other words, the deployment of trust among IoT devices is utmost important for a smooth and fair data transmission. In this regard, several approaches have been proposed in the literature, which is the main focus of this survey.

1) The first approach presented in this paper is E-Lithe. In this technique, a secret key is shared between two devices to avoid DoS attacks and therefore the security for constrained devices is increased. However, if an intruder creates multiple requests from different devices, the battery drainage becomes crucial to handle frequent computations.
2) The GTRS follows the idea of social IoT where a node sends a request towards its friends to get recommendations for the past ratings. If the best rated node is found then it is selected, otherwise, the request is forwarded to the friends-of-friends. Thus, all nodes are capable to calculate their own predictions for the best rated services. Nevertheless, the system is unable to predict the rating of a device if it has not been rated.
3) The TWGA consists of four components, i.e., i) path establishment among trust domains, ii) data forwarding to smart homes, iii) usage of public or private keys to identify the correct ID-packet, and iv) route selection through the ID-packet engine. Hence, the scheme provides useful security through public/private keys, but cannot avoid the repudiation attacks if an intruder injects false data.
4) The TBBS includes IoT enabled vehicles, traffic signals, and speed detectors to control data transfer among vehicles. Every vehicle has a default trust value that can be used in the future if needed. The TBBS may be helpful in selecting a particular route, however, this is a proposed approach and therefore it is early to predict its performance and accuracy without deployment.
5) The MAG-SIoT is based on four inter-relationships, i.e., (a) ownership object relationship, (b) co-location object relationship, (c) parental object relationship, and (d) social object relationship. The trust in this model is calculated on the basis of trust metrics, which include social relationships and the context in which the relationship is communicated. Therefore, this model is suitable to establish the relationship based on nodes' affinity, but it is inappropriate when the number of attributes increases.
6) The ATES is designed to calculate both personal and non-personal trust values of IoT devices. The personal trust value of a device is calculated in three ways, i.e., i) the current situation of a device, ii) from the experience history of a device, and iii) through M5 tree regression model. Whereas the non-personal trust value is computed with the help of other users' experience. Thus, the model can produce the ideal trust value in case of first time interaction with other devices. However, the accuracy of results depends on more number of situational characteristics.
7) In the TMSMD model, the trust is maintained at each layer of the network. The Physical layer provides data integrity and privacy, while the Application layer keeps the confidentiality of services. This model develops trust by reducing overhead and uses a public key to protect data. However, it consumes maximum power because it uses the public key cryptography, which is based on the integer factorization.
8) The DTMS is based on a distributed mechanism to provide various services in the IoT environment. The trust value of each node is calculated on direct observations. The service provided by each node has a reward if it is provided on time, and a penalty if it is not provided to the nodes. This model performs well to evaluate selective attacks in a trust management model, but it increases the chances of Bad-Mouthing attacks.
9) The SMA is designed to provide an automatic method to identity IoT devices, calculate their semantic

TABLE 1. Summary of trust management techniques.

Technique	Contributions	Limitations
E-Lithe [26]	Enhances security for constrained devices that in turn decreases DoS attacks by sharing secret keys.	When an intruder creates multiple handshake requests from multiple nodes, then the battery drainage is a crucial problem to handle frequent computations.
GTRS [27]	Computes the effectiveness of one node on another by combining their trust and similarity.	Predicting the rating for a device is quite complicated. It is also difficult to tackle the prediction issue when searching nodes are similar to each other.
TWGA [30]	More trustworthy as domains share a device-ID, signature, and a public key for establishing a trusted-ID between a home device and a service provider.	Intruders can inject false data and make repudiation attacks.
TBBS [31]	Minimizes collisions by identifying vehicles' speed and thus useful for VANET nodes to gather information for establishing an intelligent network.	It is a theoretical model and thereby difficult to predict its performance and effectiveness.
MAG-SIoT [32]	Suitable to establish the relationship based on nodes' affinity.	Inappropriate when the number of attributes increases as it expands the affinity matrix.
ATES [37]	Provides an ideal trust value in case of first time interaction with a device.	It is difficult to predict the accuracy of results from a less number of situational characteristics.
TMSMD [41]	Protects data using public key approach and develops trust by reducing computational overhead.	The public key cryptography is adopted which is based on the integer factorization, thus, consumes maximum power.
DTMS [45]	Performs well to evaluate selective attacks in a trust management model.	Increases the chances of Bad-Mouthing attacks, thus, introduces security-related issues and reduces trustworthiness.
SMA [15]	More trustworthy as it extracts text and numerical data from IoT devices through the network.	Increases computational overhead as it uses textual and numerical information for the discovery of resources and the calculation of trust score.
DTEB [51]	Transparent and immutable for recording transactions.	Neglects users' privacy, which is the main concern while dealing with data exchange in the IoT environment.
ABAC [56]	Provides secure authorization, augments scalability, and expedites the process of decision making.	Difficult to predict its trustworthiness in case one node interacts with several at a time.
ATBP [20]	Useful to decide the best route so as to avoid traffic congestion and accidents, and provide safe and smooth drive.	The travel map guide application may hinder in high dynamic situations.
DCTEPF [55]	Useful to filter out inappropriate information.	Not suitable for handling contextual data for trust predictions.
CEP [61]	The model is based on mutual trust of two users, therefore, avoids On-Off and collision attacks.	Fails to provide support in a scenario where a high mobility is involved
MAPE-K [72]	Contributes in increasing the dynamic trust management level through a self-adaptation scheme.	Fails when the service disrupts; faces problems in handling malicious nodes.
TDFDS [74]	Provides trust for online integrated and distributed applications; addresses security threats for applications, which have different nature of problems.	Addresses only limited risk vulnerabilities and does not offer security mechanisms for administrators.
CBSTM-IoT [21]	It is an adaptive model that can adjust itself according to the behavioral pattern changes and therefore does not depend on specific nodes or peers.	Malicious nodes can sometimes allocate higher trust values to other nodes, which ultimately decreases the accuracy and performance of trust calculations.
Timely Trust [85]	The GVT members use cloud servers for data storage, therefore, they can easily communicate with each other and may access data from the shared storage.	Cyber-attacks may happen as data is shared over different regions and therefore the GVT members may lose trust on smart services.
DTRM [94]	Provides protection against Bad-Mouthing, Good-Mouthing, and ballot attacks.	Cannot handle DDOS, MIM, and wormwhole attacks.
ANTs [97]	Monitors the network to identify malicious nodes.	Faces problems in dividing the network into trusted zones.
TAS-IoT [108]	The receiver legitimizes messages from the sender with no further need of authentication.	If the trust value is less than the threshold then the message is manually authenticated.
CTM-IoT [109]	Provides trustworthy communications among all IoT devices.	Its supermacy over existing techniques is vague as it is not evaluated/compared with other schemes.
TMCoI-SIOT [110]	Helps reducing storage related issues and fights against On-Off attacks.	Fails to handle Good-Mouthing and Bad-Mouthing attacks.
CTMS-SIOT [114]	Computes the social similarity among objects and thus strengthens the performance of decision-making.	Reduces the system trustworthiness as it depends on the past experience.
TMF-VSN [114]	Improves network performance by increasing the packet delivery ratio and decreasing the end-to-end delay.	The density of nodes can affect the validity of experiments.
IoT-HiTrust [117]	Achieves appropriate trust properties by considering attacks robustly in a large IoT system.	Surrenders to control intruders as it does not take into consideration the intrusion detection.

attributes, and estimate their trustworthiness. The architecture includes two parts, i.e., the smart middleware architecture and the semantic device discovery with trust evaluation. The middleware architecture calculates the trustworthiness and semantic discovery of IoT objects based on their text attributes. The SMA is more trustworthy as it extracts text and numerical data from IoT devices through the network. However, it increases computational overhead as it uses textual and numerical information for the discovery of resources and calculation of trust score.

10) The DTEB system was designed to time stamp digital documents. The system works on *Smart Contract* that uses the Blockchain architecture, which consists of four layers, i.e., Interactive layer, Management layer, Network layer, and Data layer, for exchanging data in a trusted environment. The DTEB system is transparent and immutable to record a transaction, but there are still privacy issues that make the IoT environment untrustworthy.
11) The ABAC model was proposed to keep data protected from malicious nodes. The system comprises three modules, i.e., trust evaluation, access decision, and authentication. Thus, it provides a secure authorization because the trust level changes with the behavior of nodes. However, the system accuracy cannot be predicted if one device interacts simultaneously with several other devices.
12) The ATBP is developed to allow security measures among nodes of a social network. The model adopts a trust policy that is followed by all network nodes. The ATBP suggests an application for travelers, known as *map guide*, which can be installed on a smartphone for the calculation of trust either directly or indirectly. It considers the honesty as a trust property for managing Bad-Mouthing attacks. The model is useful in deciding the best route so as to avoid traffic congestion and accidents, and provide safe and smooth drive. Though the travel map guide application gives an easy access to applications, but it may be hindered in high dynamic situations.
13) The DCTEPF system consists of various modules, for example, trust data access object, trust service enabler, decision making and prediction, trust agent, data repository, TrustComputation, and API, to calculate the trust. This system is useful to filter out inaccurate data. Nonetheless, it is not helpful to handle contextual information for trust predictions.
14) The TrustCEP is divided into two parts: The producer and the consumer, which are connected through the operator graph. This model is based on mutual trust between two users. Every user tries to find neighboring users and looks into their trust vectors. If there are no neighboring users then the graph is initiated on their own device and requests are placed for collaborating placement requests. However, it fails to provide support in a scenario with a higher mobility.
15) The MAPE-K approach was proposed to handle the dynamic environment. In the IoT Cloud environment, this scheme is cooperative as it provides facility to tackle malicious recommendations from other nodes. For quick response, the idea of distributed trust agents is used in the MAPE-K feedback loop. The proposed model helps increasing the dynamic trust management level through a self-adaptation method. Yet, the problem occurs if data attributes increase from the threshold.
16) The TDFDS model consists of four modules that define various variables of trust, i.e., customer, business requirement, and technology. The environment variable involves technological attributes as well as social, cultural, and religious factors. The customer variable includes human intelligence and their physical abilities. The business requirement variable includes attributes that affect the trust. And the technology attribute maintains system's security and usability. The primary purpose of TDFDS is to provide trust for online integrated and distributed applications, but during application running, it does not provide security for administrators.
17) The CBSTM-IoT is designed for the nodes' collaboration and to limit interactions of suspicious devices. In this model, if the relationship value is high, it indicates a higher trust. As the CBSTM-IoT model does not depend on specific nodes and peers, malicious nodes may allocate higher trust values to other nodes as indirect recommendations.
18) The *Timely Trust* framework identifies the demand of IoT in GVTs and tells that how the swift trust formation in GVTs is affected by different cultures. GVTs have common shared objectives on which they work across geographical boundaries and depends on technology such as computers to communicate. They do not have any previous working record with each other and also have cultural differences. Due to embedded IoT concepts, the GVT members can easily communicate through video calls or voice messages. However, the sharing of data on remote servers over different regions increases the chances of cyber-attacks.
19) The DTRM focuses on distributed environment to make IoT devices capable of handling processing. The model also proposes different levels of security, which are suitable for sensitive devices in the IoT environment. It keeps record of all devices and manages them according to their requirements. It provides protection against Bad-Mouthing, Good-Mouthing, and ballot attacks, but fails to handle some attacks, such as DDOS, MIM, and wormwhole.
20) The ANTs divides the network into trust zones for checking new joining nodes and reconfigures the existing trust zones. The reconfiguration of trust zones helps restricting remote communications and safeguarding

the network from several kinds of attacks. In the ANTs, the ED becomes part of the network and allows all nodes to communicate over a secure channel using SHGW. The SHGW works as a monitoring device to identify and exclude malicious nodes from the network. However, scheming suitable policies and procedures to put EDs into the trusted zones is a challenging issue.

21) In the TAS-IoT model, nodes are divided into two categories, i.e., legitimate nodes and non-legitimate nodes. The legitimate node appends on authenticator for authenticating messages. It prevents non-legitimate nodes to post false messages in the network and therefore reduces power consumption by authenticating data at its origin. A trust value is associated with each node on the basis of observations, experience, and recommendations. After the trust value is calculated, an adaptive function is used to decide if a message needs authentication.
22) The CTM-IoT is designed for reliable information sharing among IoT nodes. The IoT network is divided into different clusters, where each cluster includes a trust manager, i.e., a master node. The model also comprises a super node which stores trusted data of all master and cluster nodes in the central repository. In addition, the super node also monitors traffic and trust management among all IoT devices. Moreover, it shares data packets between the master node and cluster nodes, and the IoT applications and the master node. This model can achieve the primary goal of trust management among IoT devices, however, without comparison with other schemes, it is difficult to predict its supremacy over the existing available techniques.
23) The TMCoI-SIOT is designed to integrate various characteristics of trust on the basis of direct and indirect evaluations. The proposed architecture employs the idea of clustering and divides nodes into communities on the basis of interest, where the network consists of an SIoT server, nodes that are clustered together as a community, and a trust administrator for security management. If a node needs to join the network, the SIoT server authenticates it. After the authentication, the node may join the community of its own interest or either it can start creating its own community. The TMCoI-SIOT helps to reduce challenges associated with memory storage, however, it cannot eliminate Bad-Mouthing and Good-Mouthing attacks.
24) The CTMS-SIOT is designed to consider dynamic trust values together with a relative context in different tasks. This model is based on computational complexity, where a node's life time decreases because of information caching in a decentralized architecture. The CTMS-SIOT includes two modules, which are responsible for contextual trust and reputation. A trust request from a user activates the discovery mechanism, where lack of history in the local trust table compels a user to send a trust request query to the server. As the server receives a request, the entity selection process is initiated based on the past experience. The CTMS-SIOT is used to compute social similarities between the requester and the selected node. The model provides a dynamic environment through effective services, but it reduces the system trustworthiness.
25) The TMF-VSN is proposed for VSN, which includes three layers of trust for the VSN environment, i.e., GTM, DTM, and VTM. The GTM lies on the top level and holds the authentication of vehicles' profiles. The DTM holds the history, domain, and relationship profiles of each individual vehicle. While the VTM is used to maintain vehicles' information. The proposed model includes four modules, i.e., friend trust, neighbor trust, global trust, and history trust modules for the trust evaluation. The system can improve the performance of network by increasing the packet delivery ratio, but the validity of experiments may be affected due to nodes' density.
26) In the IoT-HiTrust, the trustworthiness of all IoT devices is calculated by a cloud in the region of cloudlets. The system is divided into three layers, i.e., the cloud layer, the cloudlet layer, and the device layer. At the cloud service level, each IoT device has a unique identity, which is used to manage users' data. The home cloud server of a user remains the same, however, its VM may be shifted from one point to the other. In case, each owner has multiple devices, then all devices are mapped to the owner home cloud. Devices' requests and replies are communicated only inside their cloudlet regions together with their stored information. If the Internet connection is terminated then a cloudlet replies user queries inside the region with a disconnection mode. Furthermore, if a user moves from one cloudlet to the other then it is removed from the previous cloudlet and is registered in the new one. The proposed model achieves an appropriate trust in a large IoT system, but it does not succeed to control intruders as it ignores the intrusion detection.

IV. CONCLUSION

IoT allows the concept of connecting billions of tiny devices to retrieve and share information regarding numerous applications, such as healthcare, environment, and industries among others. In contrast, IoT has unproven characteristics (for example, security, privacy, and trust), which are crucial in some environments such as VANETs. This paper surveys trust management techniques designed for the Internet of Things (IoT). On the basis of comprehensive analysis of trust management, relevant techniques are classified and their contributions and limitations are presented. We expect that this survey will be effective for the IoT research community, working on trust management, to comprehend the viewpoints and issues that IoT faces in trust administration.

REFERENCES

[1] N. C. Luong, D. T. Hoang, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Data collection and wireless communication in Internet of Things (IoT) using economic analysis and pricing models: A survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 4, pp. 2546–2590, 4th Quart., 2016.

[2] R. Minerva, A. Biru, and D. Rotondi, "Towards a definition of the Internet of Things (IoT)," Tech. Rep., 2015. [Online]. Available: https://iot.ieee.org/images/files/pdf/IEEE_IoT_Towards_Definition_Internet_of_Things_Revision1_27MAY15.pdf

[3] A. Gharaibeh *et al.*, "Smart cities: A survey on data management, security, and enabling technologies," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2456–2501, 4th Quart., 2017.

[4] G. Acampora, D. J. Cook, P. Rashidi, and A. V. Vasilakos, "A survey on ambient intelligence in healthcare," *Proc. IEEE*, vol. 101, no. 12, pp. 2470–2494, Dec. 2013.

[5] S. Pellicer, G. Santa, A. L. Bleda, R. Maestre, A. J. Jara, and A. G. Skarmeta, "A global perspective of smart cities: A survey," in *Proc. 7th Int. Conf. Innov. Mobile Internet Services Ubiquitous Comput. (IMIS)*, 2013, pp. 439–444.

[6] M. Elsaadany, A. Ali, and W. Hamouda, "Cellular LTE-A technologies for the future Internet-of-Things: Physical layer features and challenges," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2544–2572, 4th Quart., 2017.

[7] C.-W. Tsai *et al.*, "Data mining for Internet of Things: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 77–97, 1st Quart., 2014.

[8] D. Georgakopoulos, P. P. Jayaraman, M. Fazia, M. Villari, and R. Ranjan, "Internet of Things and edge cloud computing roadmap for manufacturing," *IEEE Cloud Comput.*, vol. 3, no. 4, pp. 66–73, Jul./Aug. 2016.

[9] N. Zhao, F. R. Yu, H. Sun, A. Nallanathan, and H. Yin, "A novel interference alignment scheme based on sequential antenna switching in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 10, pp. 5008–5021, Oct. 2013.

[10] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generat. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.

[11] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the Internet of Things: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 414–454, 1st Quart., 2014.

[12] M. Nitti, V. Pilloni, G. Colistra, and L. Atzori, "The virtual object as a major element of the Internet of Things: A survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1228–1240, 2nd Quart., 2015.

[13] S. Verma, Y. Kawamoto, Z. M. Fadlullah, H. Nishiyama, and N. Kato, "A survey on network methodologies for real-time analytics of massive IoT data and open research issues," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1457–1477, 3rd Quart., 2017.

[14] Z. Yan, P. Zhang, and A. V. Vasilakos, "A survey on trust management for Internet of Things," *J. Netw. Comput. Appl.*, vol. 42, pp. 120–134, Jun. 2014.

[15] J. Caminha, A. Perkusich, and M. Perkusich, "A smart middleware to perform semantic discovery and trust evaluation for the Internet of Things," in *Proc. 15th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2018, pp. 1–2.

[16] L. C. C. De Biase, P. C. Calcina-Ccori, F. S. C. Silva, and M. K. Zuffo, "The semantic mediation for the swarm: An adaptable and organic solution for the Internet of Things," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2017, pp. 78–79.

[17] M. Nitti, R. Girau, and L. Atzori, "Trustworthiness management in the social Internet of Things," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1253–1266, May 2014.

[18] J. Guo, I.-R. Chen, and J. J. P. Tsai, "A survey of trust computation models for service management in Internet of Things systems," *Comput. Commun.*, vol. 97, pp. 1–14, Jan. 2017.

[19] S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini, "Security, privacy and trust in Internet of Things: The road ahead," *Comput. Netw.*, vol. 76, pp. 146–164, Jan. 2015.

[20] M. N. Vidya and K. S. N. Prasad, "Adaptation trust based protocol for IoT using smartphones in social media: Travel map guide," in *Proc. 2nd Int. Conf. Appl. Theor. Comput. Commun. Technol. (iCATccT)*, 2016, pp. 109–113.

[21] S. E. A. Rafey, A. Abdel-Hamid, and M. A. El-Nasr, "CBSTM-IoT: Context-based social trust model for the Internet of Things," in *Proc. Int. Conf. Sel. Topics Mobile Wireless Netw. (MoWNeT)*, 2016, pp. 1–8.

[22] J. Heuer, J. Hund, and O. Pfaff, "Toward the Web of Things: Applying Web technologies to the physical world," *Computer*, vol. 48, no. 5, pp. 34–42, 2015.

[23] N. Modadugu and E. Rescorla, "The design and implementation of datagram TLS," in *Proc. NDSS*, 2004, pp. 1–13.

[24] S. Raza, H. Shafagh, K. Hewage, R. Hummen, and T. Voigt, "Lithe: Lightweight secure CoAP for the Internet of Things," *IEEE Sensors J.*, vol. 13, no. 10, pp. 3711–3720, Oct. 2013.

[25] P. Kasinathan, C. Pastrone, M. A. Spirito, and M. Vinkovits, "Denial-of-service detection in 6LoWPAN based Internet of Things," in *Proc. IEEE 9th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Oct. 2013, pp. 600–607.

[26] A. Haroon, S. Akram, M. A. Shah, and A. Wahid, "E-Lithe: A lightweight secure DTLS for IoT," in *Proc. IEEE 86th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2017, pp. 1–5.

[27] N. S. Nizamkari, "A graph-based trust-enhanced recommender system for service selection in IoT," in *Proc. Int. Conf. Inventive Syst. Control (ICISC)*, 2017, pp. 1–5.

[28] L. Atzori, A. Iera, and G. Morabito, "SIoT: Giving a social structure to the Internet of Things," *IEEE Commun. Lett.*, vol. 15, no. 11, pp. 1193–1195, Nov. 2011.

[29] J. O'Donovan and B. Smyth, "Trust in recommender systems," in *Proc. 10th Int. Conf. Intell. User Interfaces*, 2005, pp. 167–174.

[30] E. Kim and C. Keum, "Trustworthy gateway system providing IoT trust domain of smart home," in *Proc. 9th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, 2017, pp. 551–553.

[31] R. Talreja, S. Sathish, K. Nenwani, and K. Saxena, "Trust and behavior based system to prevent collision in IoT enabled VANET," in *Proc. Int. Conf. Signal Process., Commun., Power Embedded Syst. (SCOPES)*, 2016, pp. 1588–1591.

[32] U. S. Premarathne, "MAG-SIoT: A multiplicative attributes graph model based trust computation method for social Internet of Things," in *Proc. IEEE Int. Conf. Ind. Inf. Syst. (ICIIS)*, Dec. 2017, pp. 1–6.

[33] L. Atzori, A. Iera, G. Morabito, and M. Nitti, "The social Internet of Things (SIoT)—When social networks meet the Internet of Things: Concept, architecture and network characterization," *Comput. Netw.*, vol. 56, no. 16, pp. 3594–3608, 2012.

[34] M. Kim and J. Leskovec, "Multiplicative attribute graph model of real-world networks," *Internet Math.*, vol. 8, nos. 1–2, pp. 113–160, 2012.

[35] A. K. Dey, "Understanding and using context," *Pers. Ubiquitous Comput.*, vol. 5, no. 1, pp. 4–7, 2001.

[36] J. Huang and D. Nicol, "A formal-semantics-based calculus of trust," *IEEE Internet Comput.*, vol. 14, no. 5, pp. 38–46, Sep./Oct. 2010.

[37] H. Son, N. Kang, B. Gwak, and D. Lee, "An adaptive IoT trust estimation scheme combining interaction history and stereotypical reputation," in *Proc. 14th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2017, pp. 349–352.

[38] G. Huerta-Canepa, S. Han, D. Lee, and B. Kim, "A place-aware stereotypical trust supporting scheme," in *Proc. 12th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom)*, Jul. 2013, pp. 821–828.

[39] J. R. Quinlan *et al.*, "Learning with continuous classes," in *Proc. 5th Austral. Joint Conf. Artif. Intell.*, Singapore, 1992, pp. 343–348.

[40] C. Burnett, T. J. Norman, and K. Sycara, "Bootstrapping trust evaluations through stereotypes," in *Proc. 9th Int. Conf. Auto. Agents Multiagent Syst.*, vol. 1, 2010, pp. 241–248.

[41] K. A. R. Rehiman and S. Veni, "A trust management model for sensor enabled mobile devices in IoT," in *Proc. Int. Conf. I-SMAC (IoT Social, Mobile, Anal. Cloud) (I-SMAC)*, 2017, pp. 807–810.

[42] R. Godha, S. Prateek, and N. Kataria, "Home automation: Access control for IoT devices," *Int. J. Sci. Res. Pub.*, vol. 4, no. 10, p. 1, 2014.

[43] K. Chelli, "Security issues in wireless sensor networks: Attacks and countermeasures," in *Proc. World Congr. Eng.*, vol. 1, 2015, pp. 1–6.

[44] W. Stallings, *Cryptography and Network Security: Principles and Practice*. Pearson Education India, 2003.

[45] C. V. L. Mendoza and J. H. Kleinschmidt, "Defense for selective attacks in the IoT with a distributed trust management scheme," in *Proc. IEEE Int. Symp. Consum. Electron. (ISCE)*, Sep. 2016, pp. 53–54.

[46] Y. B. Saied, A. Olivereau, D. Zeghlache, and M. Laurent, "Trust management system design for the Internet of Things: A context-aware and multi-service approach," *Comput. Secur.*, vol. 39, pp. 351–365, Nov. 2013.

[47] C. V. L. Mendoza and J. H. Kleinschmidt, "Mitigating on-off attacks in the Internet of Things using a distributed trust management scheme," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 11, p. 859731, 2015.

[48] F. Bao and R. Chen, "Trust management for the internet of things and its application to service composition," in *Proc. IEEE Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2012, pp. 1–6.

[49] I. Yaqoob *et al.*, "Internet of Things architecture: Recent advances, taxonomy, requirements, and open challenges," *IEEE Wireless Commun.*, vol. 24, no. 3, pp. 10–16, Jun. 2017.

[50] T. Li, Y. Liu, Y. Tian, S. Shen, and W. Mao, "A storage solution for massive IoT data based on NoSQL," in *Proc. IEEE Int. Conf. Green Comput. Commun. (GreenCom)*, Nov. 2012, pp. 50–57.

[51] Z. Huang, X. Su, Y. Zhang, C. Shi, H. Zhang, and L. Xie, "A decentralized solution for IoT data trusted exchange based-on blockchain," in *Proc. 3rd IEEE Int. Conf. Comput. Commun. (ICCC)*, Dec. 2017, pp. 1180–1184.

[52] G. Zyskind, O. Nathan, and A. Pentland, "Decentralizing privacy: Using blockchain to protect personal data," in *Proc. IEEE Secur. Privacy Workshops (SPW)*, May 2015, pp. 180–184.

[53] K. Yin, C. Shou, and Z. Cai, "A data exchange optimized approach for cloud migration," in *Proc. 4th Int. Conf. Comput. Sci. Netw. Technol. (ICCSNT)*, vol. 1, 2015, pp. 169–172.

[54] R. Beck, J. S. Czepluch, N. Lollike, and S. Malone, "Blockchain—The gateway to trust-free cryptographic transactions," in *Proc. ECIS*, 2016, pp. 1–14,

[55] U. Jayasinghe, A. Otebolaku, T.-W. Um, and G. M. Lee, "Data centric trust evaluation and prediction framework for IoT," in *Proc. ITU Kaleidoscope, Challenges Data-Driven Soc. (ITU K)*, 2017, pp. 1–7.

[56] J. Wang, H. Wang, H. Zhang, and N. Cao, "Trust and attribute-based dynamic access control model for Internet of Things," in *Proc. Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discovery (CyberC)*, 2017, pp. 342–345.

[57] U. Jayasinghe, N. B. Truong, G. M. Lee, and T.-W. Um, "RpR: A trust computation model for social Internet of Things," in *Proc. Int. IEEE Conf. Ubiquitous Intell. Comput., Adv. Trusted Comput., Scalable Comput. Commun., Cloud Big Data Comput., Internet People, Smart World Congr. (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*, Jul. 2016, pp. 930–937.

[58] N. B. Truong, H. Lee, B. Askwith, and G. M. Lee, "Toward a trust evaluation mechanism in the social Internet of Things," *Sensors*, vol. 17, no. 6, p. 1346, 2017.

[59] P. Borzymek, M. Sydow, and A. Wierzbicki, "Enriching trust prediction model in social network with user rating similarity," in *Proc. Int. Conf. Comput. Aspects Social Netw. (CASON)*, 2009, pp. 40–47.

[60] L. Atzori, A. Iera, and G. Morabito, "From 'smart objects' to 'social objects': The next evolutionary step of the Internet of Things," *IEEE Commun. Mag.*, vol. 52, no. 1, pp. 97–105, Jan. 2014.

[61] R. Dwarakanath, B. Koldehofe, Y. Bharadwaj, T. A. B. Nguyen, D. Eyers, and R. Steinmetz, "TrustCEP: Adopting a trust-based approach for distributed complex event processing," in *Proc. 18th IEEE Int. Conf. Mobile Data Manage. (MDM)*, May/Jun. 2017, pp. 30–39.

[62] G. Cugola and A. Margara, "Deployment strategies for distributed complex event processing," *Computing*, vol. 95, no. 2, pp. 129–156, 2013.

[63] B. Ottenwälder, B. Koldehofe, K. Rothermel, K. Hong, D. Lillethun, and U. Ramachandran, "MCEP: A mobility-aware complex event processing system," *ACM Trans. Internet Technol.*, vol. 14, no. 1, p. 6, 2014.

[64] O. Bello and S. Zeadally, "Intelligent device-to-device communication in the Internet of Things," *IEEE Syst. J.*, vol. 10, no. 3, pp. 1172–1182, Sep. 2016.

[65] L. Militano, G. Araniti, M. Condoluci, I. Farris, and A. Iera, "Device-to-device communications for 5G Internet of Things," *EAI Endorsed Trans. Internet Things*, vol. 15, no. 1, pp. 1–15, 2015.

[66] R. Dwarakanath, B. Koldehofe, and R. Steinmetz, "Operator migration for distributed complex event processing in device-to-device based networks," in *Proc. 3rd Workshop Middleware Context-Aware Appl. IoT*, 2016, pp. 13–18.

[67] M. L. Polla, F. Martinelli, and D. Sgandurra, "A survey on security for mobile devices," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 446–471, 1st Quart., 2013.

[68] M. Granovetter, "The strength of weak ties: A network theory revisited," *Sociol. Theory*, vol. 1, pp. 201–233, Jan. 1983.

[69] E. Gilbert and K. Karahalios, "Predicting tie strength with social media," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, 2009, pp. 211–220.

[70] R. Dwarakanath, J. Charrier, F. Englert, R. Hans, D. Stingl, and R. Steinmetz, "Analyzing the influence of instant messaging on user relationship estimation," in *Proc. IEEE Int. Conf. Mobile Services (MS)*, Jun./Jul. 2016, pp. 49–56.

[71] P. Manuel, "A trust model of cloud computing based on quality of service," *Ann. Oper. Res.*, vol. 233, no. 1, pp. 281–292, 2015.

[72] S. Namal, H. Gamaarachchi, G. MyoungLee, and T.-W. Um, "Autonomic trust management in cloud-based and highly dynamic IoT applications," in *Proc. ITU Kaleidoscope, Trust Inf. Soc.*, 2015, pp. 1–8.

[73] T. H. Noor and Q. Z. Sheng, "Trust as a service: A framework for trust management in cloud environments," in *Proc. Int. Conf. Web Inf. Syst. Eng.*, 2011, pp. 314–321.

[74] M. Dorodchi, M. Abedi, and B. Cukic, "Trust-based development framework for distributed systems and IoT," in *Proc. IEEE 40th Annu. Comput. Softw. Appl. Conf. (COMPSAC)*, vol. 2, Jun. 2016, pp. 437–442.

[75] M. Blackstock and R. Lea, "Toward interoperability in a Web of Things," in *Proc. ACM Conf. Pervas. Ubiquitous Comput. Adjunct Pub.*, 2013, pp. 1565–1574.

[76] J. Guo and I.-R. Chen, "A classification of trust computation models for service-oriented Internet of Things systems," in *Proc. IEEE Int. Conf. Services Comput. (SCC)*, Jun./Jul. 2015, pp. 324–331.

[77] J. Clarke, R. R. Castro, A. Sharma, J. Lopez, and N. Suri, "Trust & security RTD in the Internet of Things: Opportunities for international cooperation," in *Proc. 1st Int. Conf. Secur. Internet Things*, 2012, pp. 172–178.

[78] J. Granatyr, V. Botelho, O. R. Lessing, E. E. Scalabrin, J.-P. Barthès, and F. Enembreck, "Trust and reputation models for multiagent systems," *ACM Comput. Surv.*, vol. 48, no. 2, p. 27, 2015.

[79] A. Bonatti, J. De Coi, L. Sauro, and D. Olmedilla, "Protune: A framework for semantic Web policies," in *Proc. Poster Demonstration Session 7th Int. Semantic Web Conf. (ISWC)*, Karlsruhe, Germany, 2008, pp. 1–2.

[80] B. Kauer, "OSLO: Improving the security of trusted computing," in *Proc. USENIX Secur. Symp.*, 2007, pp. 229–237.

[81] Z. Fan, W. Guoqing, T. Jun, Y. Mengting, and L. Xiaoli, "Trust of hardware," in *Proc. Int. Conf. Embedded Softw. Syst. Symp. (ICESS)*, 2008, pp. 202–207.

[82] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generat. Comput. Syst.*, vol. 25, no. 6, pp. 599–616, 2009.

[83] M. Nitti, R. Girau, L. Atzori, A. Iera, and G. Morabito, "A subjective model for trustworthiness evaluation in the social Internet of Things," in *Proc. IEEE 23rd Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2012, pp. 18–23.

[84] F. G. Mármol and G. M. Pérez, "Towards pre-standardization of trust and reputation models for distributed and heterogeneous systems," *Comput. Standards Interfaces*, vol. 32, no. 4, pp. 185–196, 2010.

[85] S. A. M. Yusof, N. Zakaria, and N. A. R. Muton, "Timely trust: The use of IoT and cultural effects on swift trust formation within global virtual teams," in *Proc. 8th Int. Conf. Inf. Technol. (ICIT)*, 2017, pp. 297–303.

[86] L. L. Martins and M. C. Schilpzand, "Global virtual teams: Key developments, research gaps, and future directions," in *Research in Personnel and Human Resources Management*. Bingley, U.K.: Emerald Group Publishing, 2011, pp. 1–72.

[87] Q. Jing, A. V. Vasilakos, J. Wan, J. Lu, and D. Qiu, "Security of the Internet of Things: Perspectives and challenges," *Wireless Netw.*, vol. 20, no. 8, pp. 2481–2501, Nov. 2014.

[88] F. Mattern and C. Floerkemeier, "From the Internet of computers to the Internet of Things," in *From Active Data Management to Event-Based Systems and More*. Zurich, Switzerland: ETH Zurich, 2010, pp. 242–259.

[89] M. Abomhara and G. M. Køien, "Security and privacy in the Internet of Things: Current status and open issues," in *Proc. Int. Conf. Privacy Secur. Mobile Syst. (PRISMS)*, 2014, pp. 1–8.

[90] F. Fukuyama, *Trust: The Social Virtues and The Creation of Prosperity*. New York, NY, USA: Free Press, 1995.

[91] M. Derven, "Four drivers to enhance global virtual teams," *Ind. Commercial Training*, vol. 48, no. 1, pp. 1–8, 2016.

[92] C. Cadwalladr and E. Graham-Harrison. (Mar. 2018). Revealed: 50 million Facebook profiles harvested for Cambridge analytica in major data breach. The Guardian. [Online]. Available: https://www.theguardian.com/news/2018/mar/17/cambridge-analytica-facebook-influence-us-election

[93] D. Chen, G. Chang, D. Sun, J. Li, J. Jia, and X. Wang, "TRM-IoT: A trust management model based on fuzzy reputation for Internet of Things," *Comput. Sci. Inf. Syst.*, vol. 8, no. 4, pp. 1207–1228, 2011.

[94] S. Asiri and A. Miri, "An IoT trust and reputation model based on recommender systems," in *Proc. 14th Annu. Conf. Privacy, Secur. Trust (PST)*, 2016, pp. 561–568.

[95] F. Bao and I.-R. Chen, "Dynamic trust management for Internet of Things applications," in *Proc. Int. Workshop Self-Aware Internet Things*, 2012, pp. 1–6.

[96] N. Aberomand, "Network intrusion detection classification using optimized probabilistic neural network," in *Proc. 3rd Int. Conf. Comput. Supported Edu. (COSUE)*, 2015, pp. 108–110.

[97] A. Wall, H. Raddatz, M. Rethfeldt, P. Danielis, and D. Timmermann, "ANTs: Application-driven network trust zones on MAC layer in smart buildings," in *Proc. 15th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2018, pp. 1–2.

[98] Federal Office for Information Security. (Mar. 2014). *Protection Profile for the Gateway of a Smart Metering System (Smart Meter Gateway PP)*. Accessed: May 15, 2018. [Online]. Available: https://www.commoncriteriaportal.org/files/ppfiles/pp0073b_pdf.pdf

[99] *IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments*, IEEE Standard 802.11-2010, 2010.

[100] F. Bersani and H. Tschofenig. (2007). *The EAP-PSK Protocol: A Pre-Shared Key Extensible Authentication Protocol (EAP) Method*. [Online]. Available: https://www.rfc-editor.org/rfc/rfc4764.txt

[101] D. Harkins, "Simultaneous authentication of equals: A secure, password-based key exchange for mesh networks," in *Proc. 2nd Int. Conf. Sensor Technol. Appl. (SENSORCOMM)*, 2008, pp. 839–844.

[102] Z. Shelby, K. Hartke, and C. Bormann, *The Constrained Application Protocol (COAP)*, document RFC 7252, Jun. 2014. [Online]. Available: https://tools.ietf.org/html/rfc7252

[103] E. Yuan, N. Esfahani, and S. Malek, "A systematic survey of self-protecting software systems," *ACM Trans. Auton. Adapt. Syst.*, vol. 8, no. 4, p. 17, 2014.

[104] M. Hamdi and H. Abie, "Game-based adaptive security in the Internet of Things for eHealth," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 920–925.

[105] E. K. Wang, T.-Y. Wu, C.-M. Chen, Y. Ye, Z. Zhang, and F. Zou, "MDPAS: Markov decision process based adaptive security for sensors in Internet of Things," in *Genetic and Evolutionary Computing*. Cham, Switzerland: Springer, 2015, pp. 389–397.

[106] A. V. Taddeo, M. Mura, and A. Ferrante, "QoS and security in energy-harvesting wireless sensor networks," in *Proc. Int. Conf. Secur. Cryptogr. (SECRYPT)*, 2010, pp. 1–10.

[107] A. Di Mauro, X. Fafoutis, and N. Dragoni, "Adaptive security in ODMAC for multihop energy harvesting wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 4, p. 760302, 2015.

[108] H. Hellaoui, A. Bouabdallah, and M. Koudil, "TAS-IoT: Trust-based adaptive security in the IoT," in *Proc. IEEE 41st Conf. Local Comput. Netw. (LCN)*, Nov. 2016, pp. 599–602.

[109] M. D. Alshehri and F. K. Hussain, "A centralized trust management mechanism for the Internet of Things (CTM-IoT)," in *Proc. Int. Conf. Broadband Wireless Comput., Commun. Appl.* Cham, Switzerland: Springer, 2017, pp. 533–543.

[110] O. B. Abderrahim, M. H. Elhdhili, and L. Saidane, "TMCoi-SIOT: A trust management system based on communities of interest for the social Internet of Things," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, 2017, pp. 747–752.

[111] F. Bao, R. Chen, and J. Guo, "Scalable, adaptive and survivable trust management for community of interest based internet of things systems," in *Proc. IEEE 11th Int. Symp. Auto. Decentralized Syst. (ISADS)*, Mar. 2013, pp. 1–7.

[112] I.-R. Chen, J. Guo, and F. Bao, "Trust management for SOA-based IoT and its application to service composition," *IEEE Trans. Services Comput.*, vol. 9, no. 3, pp. 482–495, May/Jun. 2016.

[113] G. Lize, W. Jingpei, and S. Bin, "Trust management mechanism for Internet of Things," *China Commun.*, vol. 11, no. 2, pp. 148–156, 2014.

[114] O. B. Abderrahim, M. H. Elhedhili, and L. Saidane, "CTMS-SIOT: A context-based trust management system for the social Internet of Things," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, 2017, pp. 1903–1908.

[115] J. R. Quinlan, *C4.5: Programs for Machine Learning*. Amsterdam, The Netherlands: Elsevier, 2014.

[116] B. Lin, X. Chen, and L. Wang, "A cloud-based trust evaluation scheme using a vehicular social network environment," in *Proc. 24th Asia–Pacific Softw. Eng. Conf. (APSEC)*, 2017, pp. 120–129.

[117] J. Guo, I.-R. Chen, and J. J. P. Tsai, "A mobile cloud hierarchical trust management protocol for IoT systems," in *Proc. 5th IEEE Int. Conf. Mobile Cloud Comput., Services, Eng. (MobileCloud)*, Apr. 2017, pp. 125–130.

[118] M. Satyanarayanan, G. Lewis, E. Morris, S. Simanta, J. Boleng, and K. Ha, "The role of cloudlets in hostile environments," *IEEE Pervas. Comput.*, vol. 12, no. 4, pp. 40–49, Oct. 2013.

[119] R. Mitchell and I.-R. Chen, "Adaptive intrusion detection of malicious unmanned air vehicles using behavior rule specifications," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 5, pp. 593–604, May 2014.

IKRAM UD DIN (S'15–SM'18) received the M.Sc. degree in computer science and the M.S. degree in computer networking from the Department of Computer Science, University of Peshawar, Pakistan, in 2006 and 2011, respectively, and the Ph.D. degree in computer science from the School of Computing, Universiti Utara Malaysia (UUM). He was a member of the InterNetWorks Research Laboratory, UUM, from 2014 to 2016. He was also the IEEE UUM Student Branch Professional Chair. He is currently a Lecturer with the Department of Computer Science, The University of Haripur, Pakistan. He has 10 years of teaching and research experience in different universities/organizations. His current research interests include resource management and traffic control in wired and wireless networks, cloud computing, traffic measurement and analysis for monitoring quality of service, mobility and cache management in information-centric networking, privacy preserving information lookup systems for information-centric architectures, and access control mechanisms for distributed content storage.

MOHSEN GUIZANI (S'85–M'89–SM'99–F'09) received the bachelor's (Hons.) and master's degrees in electrical engineering and the master's and Ph.D. degrees in computer engineering from Syracuse University, Syracuse, NY, USA, in 1984, 1986, 1987, and 1990, respectively. He was a Professor and the ECE Department Chair with the University of Idaho, USA, an Associate Vice President of Graduate Studies with Qatar University, Qatar, the Chair of the Computer Science Department, Western Michigan University, and the Chair of the Computer Science Department, University of West Florida. He held academic positions with the University of Missouri-Kansas City, the University of Colorado-Boulder, Syracuse University, and Kuwait University. He is currently a Professor with the College of Engineering, Qatar University. He has authored nine books and more than 600 publications in refereed journals and conferences. His research interests include wireless communications and mobile computing, computer networks, mobile cloud computing, security, and smart grid. He was selected as the Best Teaching Assistant for two consecutive years at Syracuse University. He received the Best Research Award from three institutions. He currently serves on the editorial boards of several international technical journals. He serves as the Founder and the Editor-in-Chief for the *Wireless Communications and Mobile Computing Journal* (Wiley). He guest edited a number of special issues in the IEEE journals and magazines. He also served as a member, the chair, and the general chair for a number of international conferences. He was the Chair of the IEEE Communications Society Wireless Technical Committee and the Chair of the TAOS Technical Committee. He served as the IEEE Computer Society Distinguished Speaker from 2003 to 2005.

BYUNG-SEO KIM (M'02–SM'17) received the B.S. degree in electrical engineering from Inha University, Incheon, South Korea, in 1998, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Florida in 2001 and 2004, respectively. His Ph.D. study was supervised by Dr. Y. Fang. Between 1997 and 1999, he was a Computer Integrated Manufacturing Engineer in advanced technology research and development with Motorola Korea Ltd., Paju, South Korea. From 2005 to 2007, he was a Senior Software Engineer in networks and enterprises with Motorola Inc., Schaumburg, IL, USA, where he focused on designing protocol and network architecture of wireless broadband mission critical communications. From 2012 to 2014, he was the Chairman with the Department of Software and Communications Engineering, Hongik University, South Korea, where he is currently a Professor. His work has appeared in around 174 publications, and he holds 25 patents. His research interests include the design and development of efficient wireless/wired networks including link-adaptable/cross-layer-based protocols, multi-protocol structures, wireless CCNs/NDNs, mobile edge computing, physical-layer design for broadband power line carrier, and resource allocation algorithms for wireless networks. He served as a member for the Sejong-City Construction Review Committee and the Ansan-City Design Advisory Board. He served as the General Chair for the General Chair of 3rd IWWCN 2017 and a TPC Member for the IEEE VTC 2014-Spring, the EAI FUTURE2016, and the ICGHIT 2016–2019 conferences. He served as a Guest Editor for Special Issues of the *International Journal of Distributed Sensor Networks* (SAGE), the IEEE ACCESS, *MDPI Sensors*, and the *Journal of the Institute of Electronics and Information Engineers*. He is an Associative Editor of the IEEE ACCESS.

SUHAIDI HASSAN (S'01–M'03–SM'08) received the B.S. degree in computer science from The State University of New York, Binghamton, NY, USA, the M.S. degree in information science (telecommunication/networks) from the University of Pittsburgh, PA, USA, and the Ph.D. degree in computing (computer networks) from the University of Leeds, U.K. In 2006, he led a task force for the establishment of the International Telecommunication Union (ITU)-Universiti Utara Malaysia (UUM) Asia Pacific Centre of Excellence for Rural ICT Development, a human resource development initiative of the ITU, which serves as the focal point for all rural ICT development initiatives across the Asia-Pacific region. He is currently a Tenure Track Professor of computing network and the Founding Chair of the InterNetWorks Research Laboratory, School of Computing, UUM. He has authored or co-authored more than 250 refereed technical publications, and successfully supervised 25 Ph.D. scholars in his research area of computer and communication networks. He was a recipient of the Swiss WKD Foundation's Young Scientist Fellowship Award at the World Knowledge Dialogue, Crans-Montana, Switzerland, in 2006. He is the Chair of the Internet Society Malaysia Chapter and is the Internet Society Fellow Alumnus to the Internet Engineering Task Force (IETF). In addition to being a speaker at a number of renowned research conferences and technical meetings, he also participates in various international fora such as ICANN meetings, Internet Governance forums, the IETF, and the IEEE meetings. He has served as a reviewer and a referee for journals and conferences as well as being an examiner for more than 100 doctoral and postgraduate scholars in his research areas. He is also an IPv6 Auditor of the Malaysian Communication and Multimedia Commission, the Malaysian ICT regulator, auditing IPv6 implementation among Malaysian leading ISPs.

MUHAMMAD KHURRAM KHAN (M'07–SM'12) is currently a Full Professor with the Center of Excellence in Information Assurance, King Saud University, Saudi Arabia. He is an Adjunct Professor with the Fujian University of Technology, China, and an Honorary Professor with IIIRC, Shenzhen Graduate School, Harbin Institute of Technology, China. He has published over 325 research papers in the journals and conferences of international repute. In addition, he is an inventor of 10 U.S./PCT patents. He has edited seven books/proceedings published by Springer-Verlag and IEEE. He is a member of the IEEE Technical Committee on Security and Privacy and the IEEE Cybersecurity Community and a fellow of the IET, U.K.; BCS, U.K.; and FTRA, South Korea. He received the Outstanding Leadership Award at the IEEE international Conference on Networks and Systems Security 2009, Australia, the Gold Medal for the Best Invention and Innovation Award at the 10th Malaysian Technology Expo 2011, Malaysia, and the Best Paper Award from the *Journal of Network and Computer Applications* (Elsevier) in 2015. He has been the Editor-in-Chief of a well-esteemed ISI-indexed international journal *Telecommunication Systems* (Springer-Verlag) since 1993 with an impact factor of 1.542 (JCR 2016), and the Founding Editor of the *The Bahria University Journal of Information and Communication Technology*. Furthermore, he is a full-time editor/associate editor of several ISI-indexed international journals/magazines, including the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, *IEEE Communications Magazine*, the *Journal of Network and Computer Applications* (Elsevier), the IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, the IEEE ACCESS, *Security and Communication Networks*, *IEEE Consumer Electronics Magazine*, the *Journal of Medical Systems* (Springer), *PLOS One*, *Computers and Electrical Engineering* (Elsevier), the *IET Wireless Sensor Systems*, *Electronic Commerce Research* (Springer), the *Journal of Computing and Informatics*, the *Journal of Information Hiding and Multimedia Signal Processing*, and the *International Journal of Biometrics* (Inderscience). He is one of the organizing chairs of more than five dozen international conferences and a member of technical committees of more than 10 dozen international conferences.