sensors

Article

Trustability for Resilient Internet of Things Services on 5G Multiple Access Edge Cloud Computing

Suleyman Uslu [1], Davinder Kaur [1], Mimoza Durresi [2] and Arjan Durresi [1,*]

[1] Department of Computer & Information Science, Indiana University Purdue University Indianapolis, Indianapolis, IN 46202, USA
[2] Department of IT, Mathematics and Statistics, European University of Tirana, 1000 Tirana, Albania
* Correspondence: adurresi@iupui.edu

Abstract: Billions of Internet of Things (IoT) devices and sensors are expected to be supported by fifth-generation (5G) wireless cellular networks. This highly connected structure is predicted to attract different and unseen types of attacks on devices, sensors, and networks that require advanced mitigation strategies and the active monitoring of the system components. Therefore, a paradigm shift is needed, from traditional prevention and detection approaches toward resilience. This study proposes a trust-based defense framework to ensure resilient IoT services on 5G multi-access edge computing (MEC) systems. This defense framework is based on the trustability metric, which is an extension of the concept of reliability and measures how much a system can be trusted to keep a given level of performance under a specific successful attack vector. Furthermore, trustability is used as a trade-off with system cost to measure the net utility of the system. Systems using multiple sensors with different levels of redundancy were tested, and the framework was shown to measure the trustability of the entire system. Furthermore, different types of attacks were simulated on an edge cloud with multiple nodes, and the trustability was compared to the capabilities of dynamic node addition for the redundancy and removal of untrusted nodes. Finally, the defense framework measured the net utility of the service, comparing the two types of edge clouds with and without the node deactivation capability. Overall, the proposed defense framework based on trustability ensures a satisfactory level of resilience for IoT on 5G MEC systems, which serves as a trade-off with an accepted cost of redundant resources under various attacks.

Keywords: trustworthy and resilient systems; trust management; internet of things sensor security; 5G multiple access edge computing security; attack mitigation

check for updates

Citation: Uslu, S.; Kaur, D.; Durresi, M.; Durresi, A. Trustability for Resilient Internet of Things Services on 5G Multiple Access Edge Cloud Computing. *Sensors* **2022**, *22*, 9905. https://doi.org/10.3390/s22249905

Academic Editor: Antonio Corradi

Received: 31 October 2022
Accepted: 13 December 2022
Published: 16 December 2022

1. Introduction

Future fifth-generation (5G) wireless cellular networks will support billions of Internet of Things (IoT) sensors and devices, including static and mobile endpoints, various robots, and self-driving cars, as illustrated in Figure 1. These devices and the related applications will attract and amplify the risk of vulnerability. 5G wireless communication technologies under development promise tremendous improvements in many areas including speed, connectivity, and reduced latency. These 5G networks can enable the movement of massive amounts of data to connect distant sensors across a critical environment, as illustrated in Figure 1. IoT on 5G will function as an integrated system with multi-access edge computing (MEC) as an extension of cloud services. IoT and MEC systems will both be under various attacks; therefore, we propose a defense framework to ensure the resilience of the entire system under attack.

A variety of customers, such as individuals employing 5G devices for personal use or large corporations and institutions, are widely using cloud computing platforms. Therefore, a wide range of applications is migrated into the cloud, including e-commerce, data storage, healthcare, gaming, and different web applications [1,2]. This allows customers to deploy and scale their services with much less effort, especially without hardware purchase requirements [3]. However, it creates new concerns, such as with security and privacy [4,5]. These are the issues that customers consider before using cloud vendors and selecting the most appropriate provider. Therefore, as cloud providers recognize these issues, they develop their services such that they would address the customers' concerns in order to attract them [1].

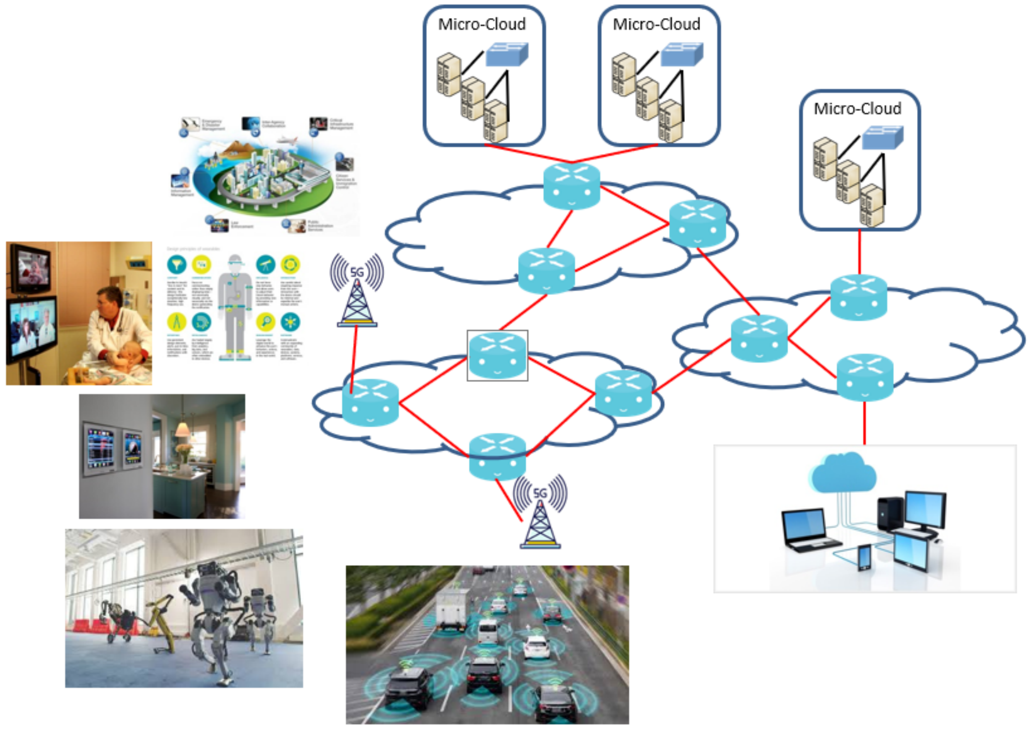

Figure 1. Internet of Things on 5G multiple-access edge computing systems.

Similarly, devices and sensors that are connected via the internet or other types of connections are being widely adopted. The speed of this adaptation process is estimated to continue increasing as the new-generation networks such as 5G spread [6,7]. These 5G networks have had many advances in wireless networking [8–11].

There are a variety of sensors in such devices, whether it be an autonomous vehicle, a car with an active safety system, a robotic vacuum, or some other IoT device [12]. For such systems, the security of the communication between the decision makers and the sensors is critical. An attack on one of the sensors could cause undesirable outcomes specific to the task [13] or simply unauthorized access to sensitive information such as healthcare data [14].

Vendors have been implementing security measures in both cloud and systems with communicating parts [5]; however, it is difficult to completely protect the whole system from attackers [15]. Faced with such new challenges, the old security model of defending the system's perimeter is no longer valid. We must assume that whatever defense mechanisms we deploy in the system will sooner or later be breached by attackers. Therefore, it is advisable to implement new techniques, including trustworthiness assessments [16], which would help the service to survive the attacks despite having to face the cost of these techniques, such as active tracking, dynamic resource allocation, and purchase of new resources. According to the Cybersecurity and Infrastructure Security Agency (CISA), the current cyberspace shifts the attention from detection and perimeter defenses to strengthening security with resilience [17,18]. A robust mechanism that ensures resilience is the deployment of redundant resources based on the assessment of the trustworthiness of the system services. Current methods may not adequately assess the trustworthiness of the systems and their components due to disproportionate heterogeneity and multi-level hierarchies.

Various studies [19–21] and surveys [22,23] have been carried out on trust management frameworks and their applications. Ruan et al. [24] proposed a measurement theory-based trust management framework to provide improved flexibility to context-dependent applications by supporting multiple formulations and a new metric: the confidence of trust. Applications of this framework include stock market prediction using Twitter data [25], trust management in environmental decision making [26–28], and the detection of crime [29], fake users [30], and damaging users [31]. These applications show the potential of utilizing a trust management framework to facilitate decision making in various fields by measuring and assessing trust.

Trust frameworks have also been proposed to be implemented in both cloud and IoT scenarios and other network-related ones, such as the scenarios that include 5G [32,33]. Ruan et al. also proposed a trust management framework for IoT [34], multi-access edge computing [35], and cloud computing platforms [36]. Furthermore, Kaur et al. [37] proposed the use of a geo-location and trust-based framework to filter out attackers in 5G social networks. These applications serve as stepping stones toward trustworthy artificial intelligence (AI) and decision making, which have been consistently promoted by researchers [38–44], governments, institutions, and organizations such as the European Union [45] and the International Organization for Standardization [46].

Park et al. [13] highlighted the significance of the security and privacy of communication and connectivity functions and proposed machine learning approaches to detect anomalies in in-vehicle networks. Furthermore, Cao et al. [47] surveyed the emerging threats in deep learning-based autonomous driving and listed different types of attacks on sensors, such as jamming and spoofing. In addition to 5G, research has been carried out and a framework has been proposed for sixth-generation (6G) networks, specifically investigating the technology's applicability and the privacy concerns in relation to unmanned aerial vehicles (UAV) [48–50]. Ullo et al. [12] also highlighted the importance of intelligent environment monitoring systems that use IoT and sensors; however, vendors and providers would need precise metrics to take the necessary actions on time in order to assess the trustworthiness of the systems.

To address the concerns about measuring the different aspects of trustworthiness, the metrics of acceptance [51] and fairness [52] were proposed to facilitate environmental decision making, an explainability metric [53] was proposed to interpret AI medical diagnosis systems, and a trustability metric [54] to assess trust in cloud computing. This paper presents an extended version of the trust management framework that includes the trustability metric, which helps to take action when an external attack or an internal event occurs in an autonomous device equipped with sensors or in a service running on the cloud. First, a sampling subsystem is explained as part of an autonomous system consisting of one decision maker and two sensors; an attack on one of the sensors is simulated, and the change in the trustability of the sensor and the entire system is shown. Then, the simulation is repeated with increased redundancy by adding another sensor. Finally, another scenario is simulated, where the extra sensor can be activated later, for instance, when the sensor lifetime is essential.

The findings illustrate the utilization of the trust management framework and the trustability metric in multiple incident scenarios within a sample cloud structure. The sample cloud consisted of three nodes, where the trust of one of the nodes declined relatively, continually, or sharply for both a short and extended period of time. It was shown that the trustability metric captures the decline in trust in the entire service. Then, additional scenarios were explored, where extra nodes could be added to each task in order to keep the service trustability high with an increased expense. Furthermore, results were shown for the cloud that had the option to remove nodes, specifically the ones with low trust. Finally, the net utility metric was illustrated to compare these two scenarios with and without the node removal option. The main contributions of this paper are as follows.

- The trustability metric was demonstrated using a sampling subsystem with a sensor activation option, where an external attack occurs on a sensor;
- Different possible outcomes of internal incident scenarios were presented in a sample cloud environment, where the trustability of the service is tracked by the framework for each scenario;
- The trustability metric captured the trustability of the service whenever the cloud architecture allowed for the addition and removal of extra nodes for each task;
- The net utility function captured the need for additional nodes and helped to decide when to remove nodes in order to optimize the utility of the service;
- Overall, this paper proposes the use of a trust management framework with a trustability metric and a net utility function on a variety of external and internal incident scenarios in order to help take timely actions to keep the service alive and optimize the utility.

The rest of the paper is organized as follows. In Section 2, the trust management framework is introduced, which is tailored to measure the trust of sensors and nodes to capture overall trustability. In Section 3, the results of utilizing the framework and how it captures trustability are presented and discussed in (i) a subsystem with sensors, where an external attack occurs to a sensor, and (ii) a sample cloud, where internal incidents happen to a node. In Section 4, findings and contributions are summarized, and future work is discussed.

2. Materials and Methods

This section presents the trust management framework and how it is adapted to capture the trustability of a system or service while considering its cost and utility.

2.1. Trust Management Framework

In [24], a measurement theory-based trust management framework was proposed for online social communities. This framework has since been proven to facilitate decision-making in multiple areas such as online social networks [25], the food-energy-water nexus [44], crime detection [29], and cancer diagnosis [53]. It is a very flexible yet robust framework that can be adapted to different scenarios to capture trust.

The framework has two main components: impression, represented by m, and confidence, represented by c. The impression is the level of trust one party shows the other, and confidence is the degree of certainty of the impression. Although different formulations are possible [24], we selected the intuitive ones, as shown in Equations (1) and (2), in order to focus on the framework. In these equations, $m^{A:B}$, $c^{A:B}$, and $r_i^{A:B}$ represent the impression, confidence, and a measurement from A to B.

$$m^{A:B} = \frac{\sum_i^N r_i^{A:B}}{N} \tag{1}$$

$$c^{A:B} = 1 - 2\sqrt{\frac{\sum_i^N (m^{A:B} - r_i^{A:B})^2}{N(N-1)}} \tag{2}$$

Trust measurements are context-dependent, which means that measurement needs to be precisely defined and specific to the context. In this study, the alternative ways of obtaining measurements [35] were combined, and predefined trust measurements were used to reflect the incidents better and to be able to concentrate on the framework and decisions. Furthermore, measurements were always normalized to be in [0–1], which caused the impression to remain in the same interval of [0–1] as an arbitrary unit.

2.2. Trust Management in Systems and Cloud

The proposed trust management framework can be adapted to scenarios where trust is assessed for parts of an autonomous system or nodes of a cloud. There are several proposed approaches [54] to measure trust, such as by measuring node flows, dividing them into incoming and outgoing, assigning different weights to such flows, and considering the trust of tasks inside of a node. Equation (3) shows that the trust of a node, m_{node}, can be measured as the weighted average of trust of the flow, m_{flow}, and tasks running on it, m_{task}. As shown in Equation (4), just as the trust of the tasks running on a node can affect its trust, the node itself can affect the trust in those tasks.

$$m_{node} = \frac{w_{flow} m_{flow} + \sum_i w_{task_i} m_{task_i}}{w_{flow} + \sum_i w_{task_i}} \tag{3}$$

$$m_{task} = \frac{w_{flow} m_{flow} + \sum_i w_{node_i} m_{node_i}}{w_{flow} + \sum_i w_{node_i}} \tag{4}$$

In this study, trust measurements were simulated for different scenarios, such as a normal condition where the trust stays the same, with the exact high measurements of around 0.9 for 10 time intervals where time has an arbitrary unit for the absence of complication. Then, scenarios where the node received lower measurements were explored, reflecting an anomaly in either the flow or the task activity, which has two types: a short-term and a continuous decline in trust. Subsequently, additional scenarios where the node received very low measurements were explored. Equation (5) represents the examples of such trust measurements, which are also shown in Figure 2.

$$\begin{aligned}
\text{Regular} &= \{0.9, 0.9, 0.9, 0.9, 0.9, 0.9, 0.9, 0.9, 0.9, 0.9\} \\
\text{Short-term decline} &= \{0.9, 0.8, 0.7, 0.6, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5\} \\
\text{Continuous decline} &= \{0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0.1\} \\
\text{Sharp short-term decline} &= \{0.9, 0.1, 0.1, 0.1, 0.9, 0.9, 0.9, 0.9, 0.9, 0.9\} \\
\text{Sharp continuous decline} &= \{0.9, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1\}
\end{aligned} \tag{5}$$

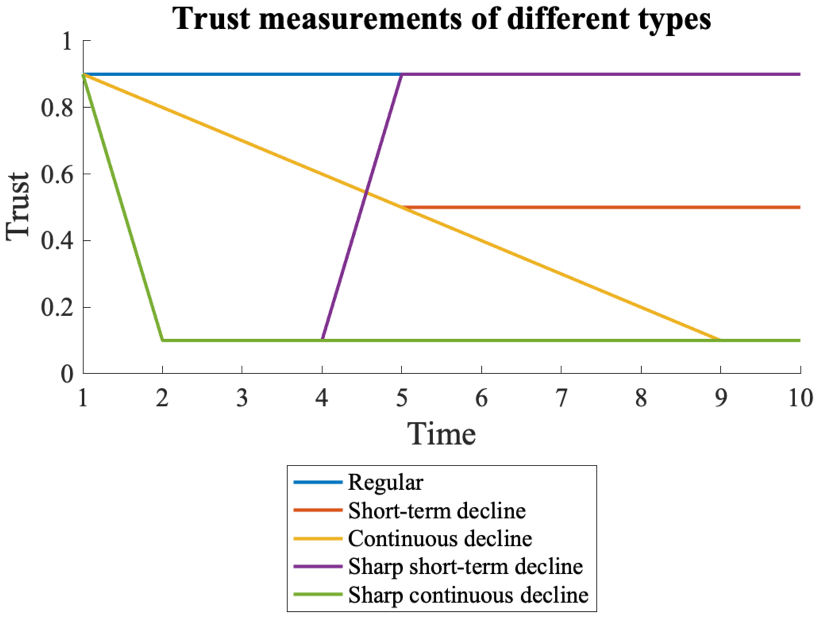

Figure 2. Trust measurements while different types of attacks occur.

2.3. Redundancy, Cost, and Utility

The trust of the entire system was also explored, whether it be an autonomous system already deployed in the field or a cloud system that could be managed later on. The overall *trustability* of the system was measured by considering the individual trust of the nodes and their hierarchy, such that the nodes that were connected in series in logical representation were all required to have high trust, whereas the nodes connected in parallel compensated for each other's abnormalities.

As discussed in [35], individual trustability was calculated using an exponential formula with two different lambdas, λ_1 and λ_2. This is because impression, m, and confidence, c, needed to be merged in order to reach a high trustability only when m and c were both high. Moreover, a threshold, ϕ, was provided for trustability to be adjusted for the application. In this study, the sample threshold used for demonstration was 0.5. In other words, the scenarios where m went below 0.5 had much lower trustability by using λ_2, whereas λ_1 was used otherwise. First, m and c were normalized and merged, as shown in Equation (6), where ϕ is the threshold. Then, trustability, τ, was calculated using the formula given in Equation (7), with the appropriate λ, which were assigned as $\lambda_1 = 4$ and $\lambda_2 = 8$. Trustability calculation is also shown in Algorithm 1.

$$\nu = \frac{2(m-\phi)c+1}{2} \tag{6}$$

$$\tau = e^{-\lambda(1-\nu)} \tag{7}$$

Algorithm 1: Trustability, τ, is calculated as an exponential function, where λ is decided by comparing the impression, m, with the threshold, ϕ.

```
Input: $m, c$
Output: $\tau$
$\lambda_1 \leftarrow 4$
$\lambda_2 \leftarrow 8$
$\phi \leftarrow 0.5$
$v \leftarrow \frac{2(m-\phi)c+1}{2}$
if $m \geq \phi$ then
    $\lambda \leftarrow \lambda_1$
end
else
    $\lambda \leftarrow \lambda_2$
end
$\tau \leftarrow e^{-\lambda(1-v)}$
```

After calculating individual trustability, the entire system's trustability was calculated. First, the trustability of the nodes that were connected in parallel was aggregated, as shown in Equation (8). Then, the transitive trustability was calculated using the formula given in Equation (9). The final value reflects the entire system's trustability, whether an active cloud system or an autonomous system that had been deployed in the field.

$$\tau_{aggr} = 1 - \Pi_i(1-\tau_i) \tag{8}$$

$$\tau_{tran} = \Pi_i \tau_{aggr_i} \tag{9}$$

Moreover, a formula to calculate the net utility of the service, including the cost of resources and the probabilities of success and failure, was developed using the trustability of the service. As shown in Equation (10), trustability, τ, was used for success, while $1-\tau$ was for failure; G represents the gain, and L denotes the loss. Then, the cost of resources was deducted, representing the sum of the cost of all nodes for the cloud scenario.

$$U_{net} = \tau G - (1-\tau)L - \sum C_{node_i} \tag{10}$$

In Section 3, trustability results are presented for the scenarios wherein such systems with different configurations experience different incidents, thus causing node and system-level trustability declines.

3. Results and Discussion

This section presents the results for two types of attacks on systems. First, the effects of external attacks on systems already deployed in the field are presented as well as the possible actions to take afterward. For instance, a sensor could lose its trustability by providing either inadequate or unreliable data. Furthermore, results and potential cautions are discussed after a decline in the trustworthiness of a service running on the cloud due to internal anomalies. In such a scenario, a service runs on multiple nodes deployed on the cloud, and a subset of the nodes loses trustability due to various internal factors such as high usage of processing power, memory, or bandwidth or a compromised task running on a node.

3.1. External Attacks

This section explores the external attacks and related factors on the sensors and subsystems of a system running in the field. In addition, the trust management system is demonstrated by capturing such conditions and facilitating decision making. In Figure 3, a sample diagram of an active safety system is shown in a vehicle that has multiple sensors and two decision-making mechanisms. In real systems, higher levels of hierarchies among decision makers and sensors are predicted [13].

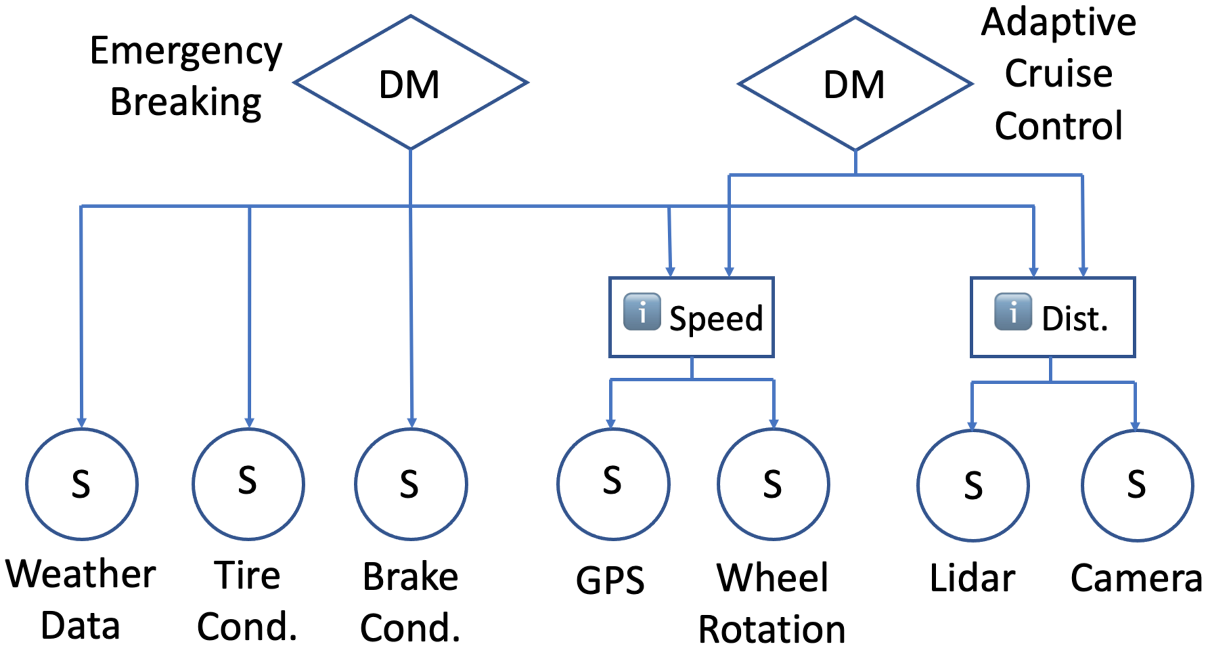

Figure 3. Sample diagram of an active safety system in a vehicle.

For example, a vehicle with an active safety system is expected to make decisions based on the information it receives from the sensors. However, such information may not always be reliable due to compromised sensors or altered sensor data. In such cases, the decision-making mechanism should be able to take the necessary actions for the safety and reliability of the decision. For instance, a dead battery in a tire pressure sensor could cause an incorrect tire condition measurement that would then affect the braking decision. Similarly, malware in a camera system that measures the distance to the vehicles in front could cause an erroneous distance measurement, which is crucial for emergency braking and adaptive cruise control.

The systems of sensors and decision-making mechanisms could quickly get complicated with multiple layers and hierarchies. We illustrate our trustworthy approach by using a simple system that can be considered a subsystem of the entire mechanism. In the first system, System-1, one decision-making element (DM) relies on two sensors, S1 and S2. It receives information from these sensors and makes a decision. This system can also be represented as a logical system, where the sensors are connected in series. Figure 4 shows the sample system and its logic representation.

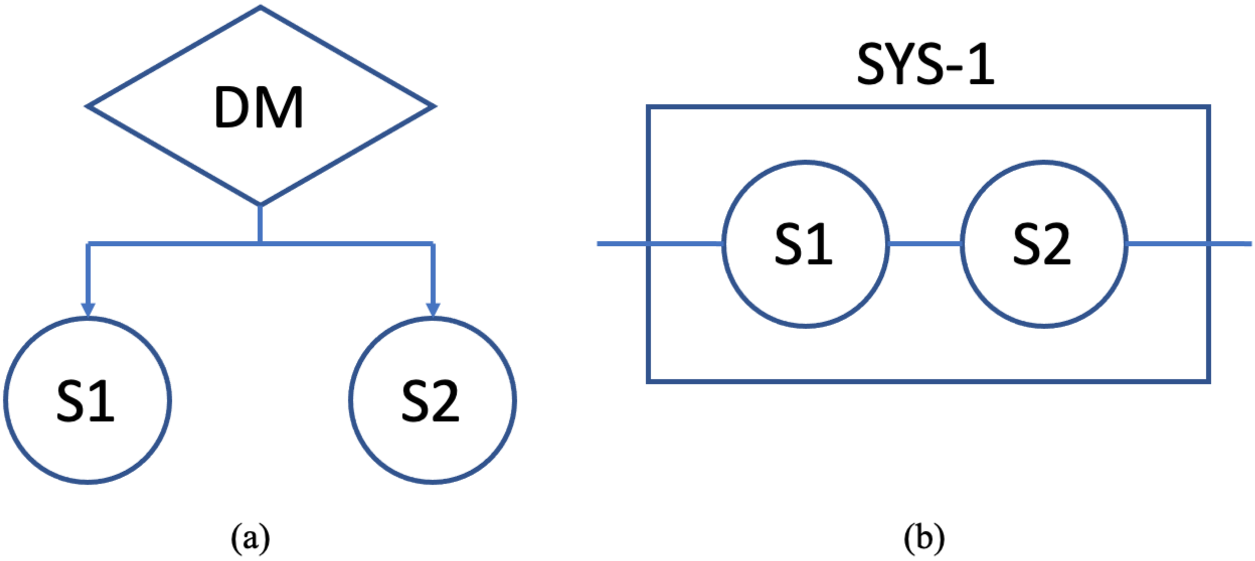

Figure 4. Sample system (System-1) with one decision maker (DM) and two sensors, S1 and S2. (**a**) System diagram as DM relies on S1 and S2. (**b**) Logic representation where the sensors are connected in series.

We explored a scenario wherein the trust of the first sensor, S1, declines. In the beginning, both sensors were assumed to have high trust, 0.90 and 0.95, respectively, because of consecutive measurements with the same values, as shown in Equation (11). However, due to a hypothetical external attack, the value of S1 became 0.1, which caused trust to decline, as shown in Figure 5. This also caused a decline in the trustability of the system, which was calculated using Equation (7).

$$\begin{aligned} S1 &= \quad \{0.9, 0.9, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1\} \\ S2 &= \{0.95, 0.95, 0.95, 0.95, 0.95, 0.95, 0.95, 0.95, 0.95, 0.95\} \end{aligned} \tag{11}$$

One option could be to include another sensor for the same task in order to overcome the adverse effects of losing the ability to make trustworthy decisions as a result of one sensor failure. The system was updated such that the DM could rely on two sensors, S1-A and S1-B, for the information that previously required its reliance only on S1. Figure 6 shows the diagram of System-2 and its logic representation, where S1-A and S1-B are connected in parallel.

When the previous scenario occurred and S1-A's trust declined, the overall trustability of System-2 did not decrease as it did in System-1. The initial trustability of System-2 was also higher than that of System-1 due to the fact that it had the additional sensor, S1-B, in the system from the beginning. Figure 7 shows the change in the overall trustability of System-2 and the trust of the sensors over time.

Another scenario is to have the additional sensor, S1-B, in the system, but to have it activated only when needed. When the trustability of the system is below a specific threshold, another sensor is activated to bring the overall trustability back to an acceptable level, as shown in Figure 8. A grace period for sensor activation was added to reflect a more realistic scenario. This scenario could happen when a sensor has a short lifetime and is only activated when the other sensor does not satisfy the requirements anymore. One drawback of such an approach is the lower initial trustability as compared to when all sensors activated, which can also be observed when Figures 5 and 7 are compared.

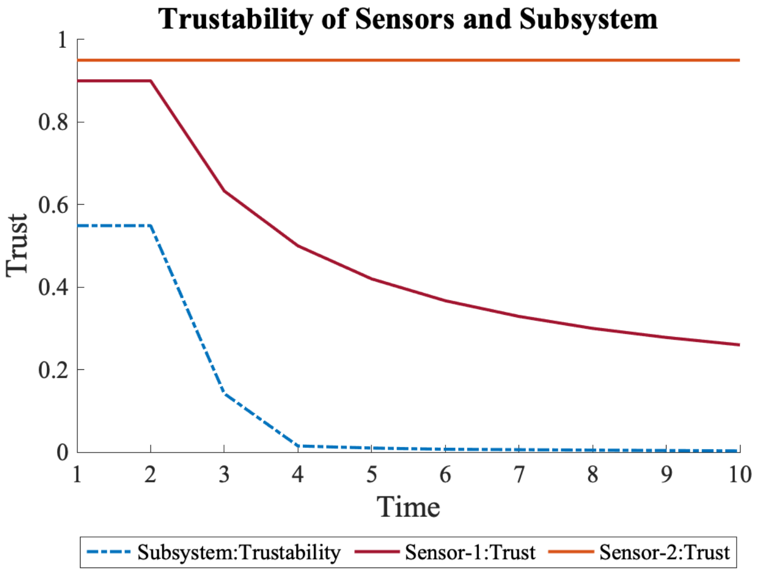

Figure 5. Trustability of System-1 and the sensors. The subsystem is susceptible to any trust fluctuation due to serially connected sensors.

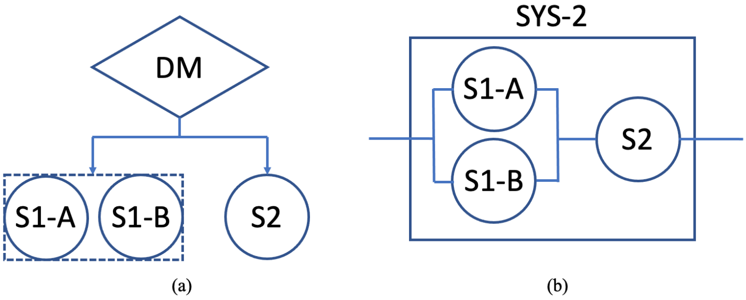

Figure 6. Sample system (System-2) with one decision maker (DM) and three sensors, S1-A, S1-B, and S2. (**a**) System diagram as DM relies on three sensors, where S1-A and S1-B are used for the same information. (**b**) Logic representation, where S1-A and S1-B are connected in parallel and are connected to S2 in series.

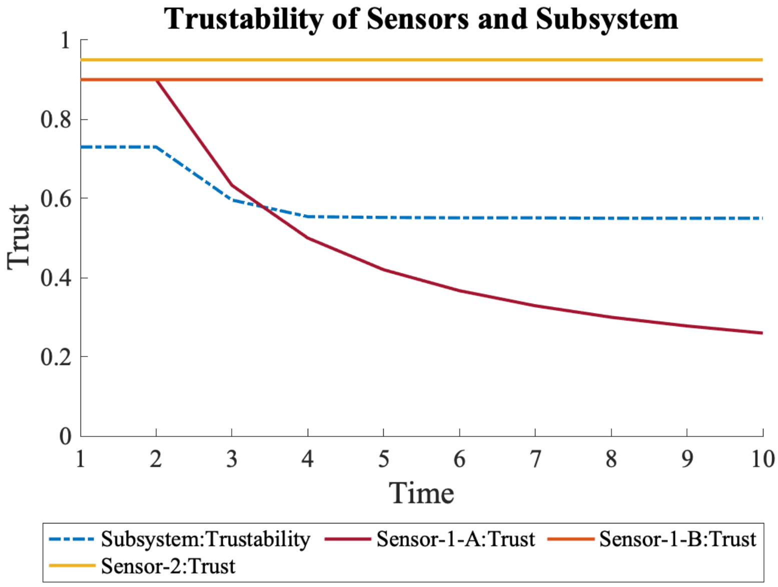

Figure 7. Trustability of System-2 and the sensors. The overall trustability of System-2 is more resistant to the trust surges of the sensors that are connected in parallel.

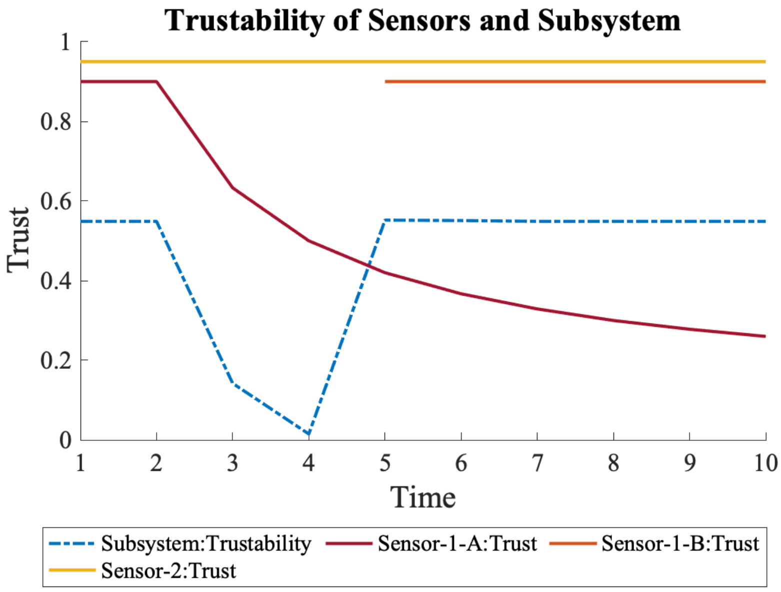

Figure 8. The overall trustability of System-3 and the sensors. S1-B is only activated after distrust of S1-A decreases the system's trustability below a threshold. Activation takes time.

3.2. Internal Incidents

This section explores the incidents affecting some cloud nodes that run a service. The trust management framework captured the decline in the trustability of the service and nodes and assisted in taking timely actions to keep the service reliable. Although today's cloud infrastructures can be very complicated and highly hierarchical, the scenarios were built on a sample cloud that had three nodes, with each running a different task.

Internal incident scenarios on cloud services differ from the system scenarios with sensors explained in Section 3.1. While the number of sensors should be decided before the device's production, cloud scenarios have more flexibility. A task can be migrated to another node, or alternative nodes can be launched. This also brings dynamic cost optimization into the picture since the nodes can be dynamically added and removed.

The sample cloud architecture consisted of three nodes connected in series, which means that the service needed three tasks deployed on different nodes. Since the reliability of the service depends on all three nodes, those nodes can be considered connected in series, as shown in Figure 9.

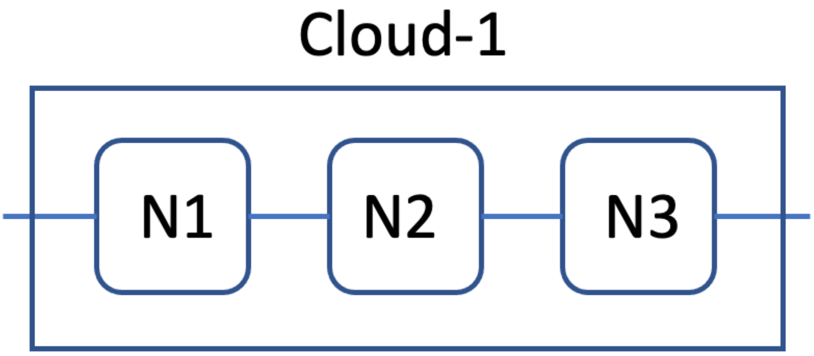

Figure 9. Sample cloud structure consisting of three nodes, each running a different task.

Four different scenarios were explored, as introduced in Equation (5), each of which caused other declines in the trustability of the service. For each scenario, only the first node, N1, was affected. In the first scenario, N1's trust measurements declined slowly until time 5 and then stayed the same, as shown in Equation (12). This caused a slow decline in the trust of N1 until time 5, which subsequently started settling gradually. However, since the nodes were connected in series, even a slight decrease in trust for one of the nodes caused a considerable decline in the trustability of the service deployed on the cloud. Figure 10 shows the change in the trustability of the service and the nodes.

$$\text{N1} = \{0.9, 0.8, 0.7, 0.6, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5\} \tag{12}$$

When the decline in trust of N1 continued, as shown in Equation (13), it placed more significant stress on the overall trustability of the service. It even decreased the overall trustability to almost zero, as shown in Figure 11, which is a sign that the system is not trustworthy anymore due to the low trust of N1.

$$\text{N1} = \{0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0.1\} \tag{13}$$

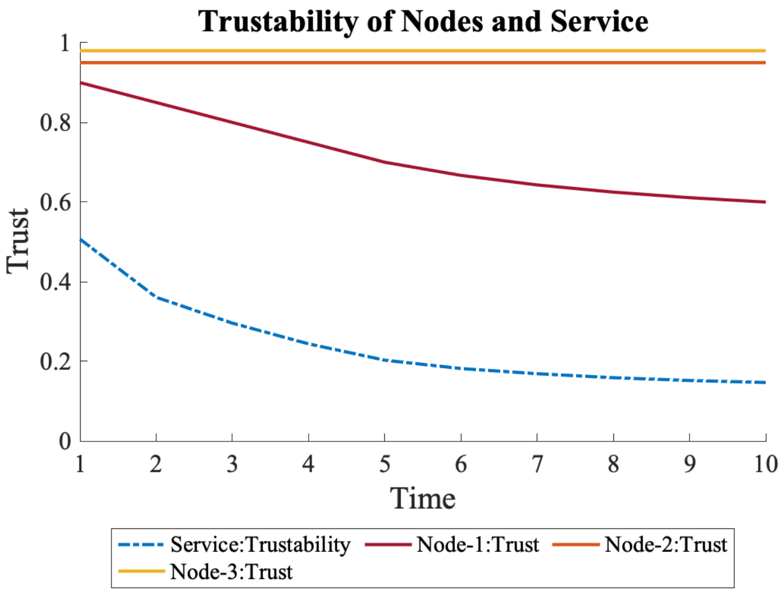

Figure 10. Trustability of a service running on three nodes on the cloud. A decline in the trust of N1 causes a decline in overall trustability.

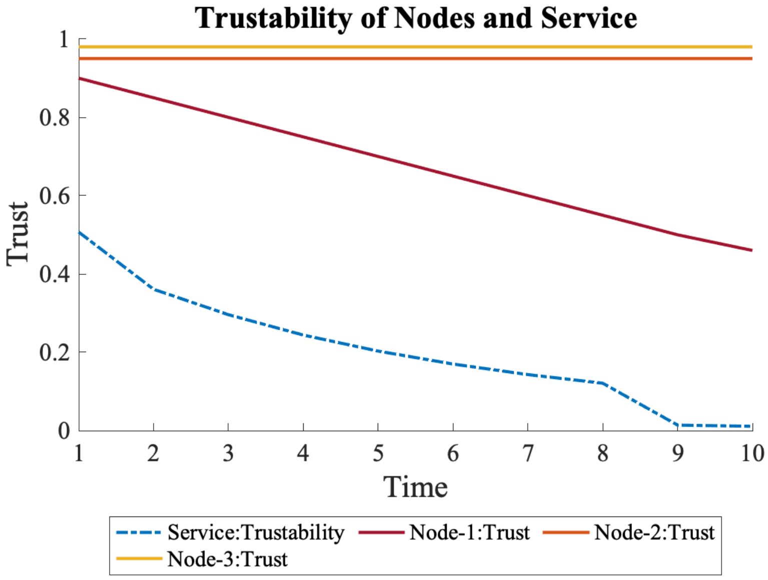

Figure 11. When trust of N1 continuously declines, its effects on the overall trustability is more severe.

Furthermore, a sudden decline in the trust measurements of N1 was explored, which returns to normal after some time, as shown in Equation (14). For example, in this case, while the regular trust measurements were at 0.9, it dropped to 0.1 at times 3 and 4 due to an internal anomaly, such as high processing power, memory, or bandwidth usage. As shown in Figure 12, the trust of N1 gradually recovered; however, the overall trustability dropped close to zero at time 4 and recovered slowly due to historic trust measurements.

$$N1 = \{0.9, 0.1, 0.1, 0.1, 0.9, 0.9, 0.9, 0.9, 0.9, 0.9\} \tag{14}$$

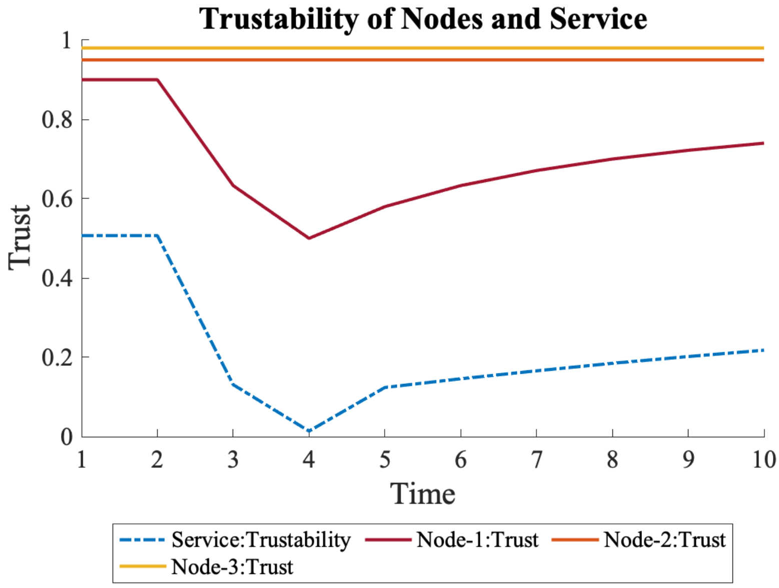

Figure 12. If the trust declines rapidly for a short period of time, overall trustability also declines but does not recover quickly.

In the final scenario, the sudden decline in the trust of N1 was set to be permanent. When the trust of N1 did not recover, as shown in Equation (15), the trustability metric did not recover either, as shown in Figure 13. As in the previous scenario, it declined close to zero, indicating the service's low trustability.

$$N1 = \{0.9, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1\} \tag{15}$$

In response to the decline in the trustability of the service, some measures can be taken. One example is to use alternative nodes for each task. It provides the service to utilize both nodes for the specific task. However, this option comes with a cost since the service provider would be charged for each node operated. As we mentioned previously, a more sophisticated approach would be to use the nodes when needed.

As shown in Figure 14, Cloud-2 can add a node for each task and remove them on demand. A new scenario was explored where the trust of each primary node, N1, N2, and N3, declined gradually, starting at different times. After a grace period of 1, a new node was launched for each task to increase the overall trustability.

Once N1's trust started to decline at time 2, the overall trustability of the service also decreased. After a grace period, N4 was activated at time 4, which increased the trustability of the service again. The new trustability was more than the initial trustability since N1 was still active despite the lower trust. At the same time, N2's trust started to decrease, which was compensated by launching N5 at time 6. Finally, N3's trust started to decline, and N6 was launched.

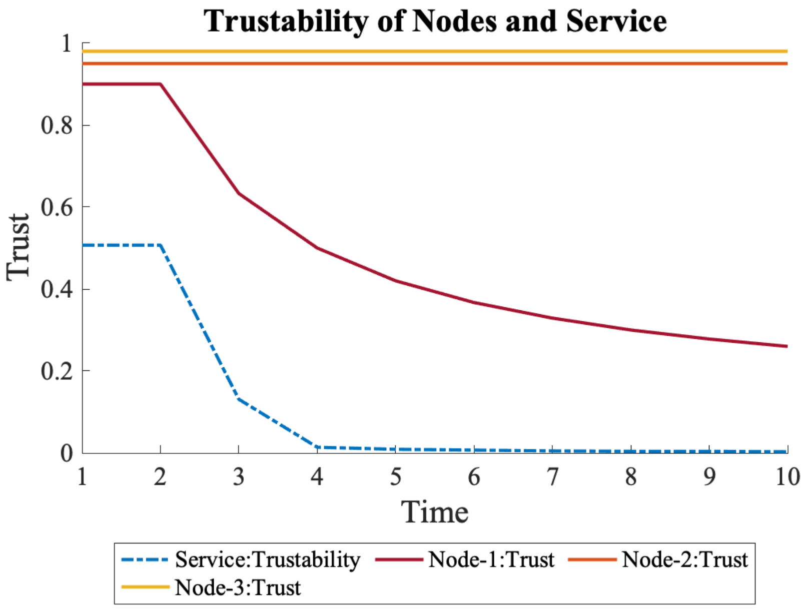

Figure 13. A sudden drop in trust measurements causes a continuous decline in the trust of N1. This reflects a more dramatic decline in the overall trustability of the service.

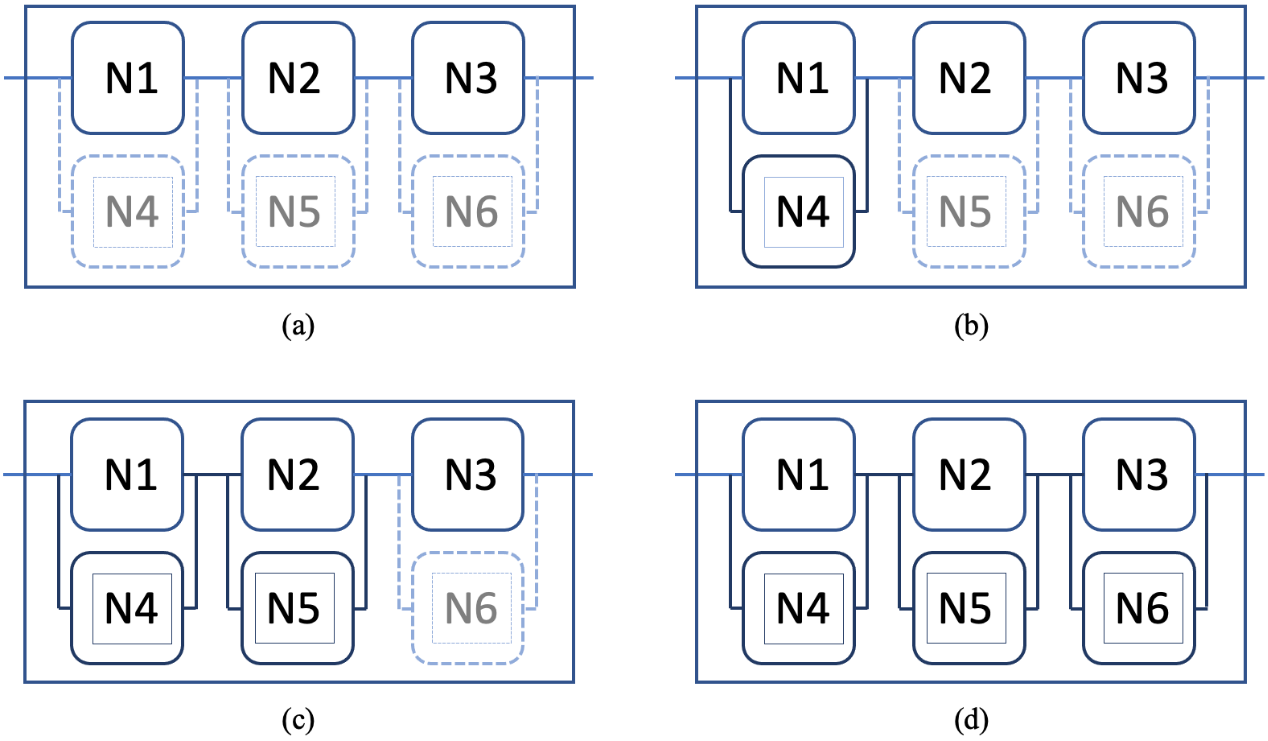

Figure 14. Cloud structure updated with additional nodes for each task. (**a**) The original Cloud-2 with N1, N2, and N3, including the options to add N4, N5, and N6. (**b**) N4 is added after N1's trust declines. (**c**) N5 is added after N2's trust declines. (**d**) N6 is added after N3's trust declines.

As seen in Figure 15, the overall trustability surpassed the initial value since the old nodes were still active and had nonzero trust values. Another conclusion is that the system became more resilient to individual trust declines once the nodes started having alternatives, such as having additional nodes that were connected in parallel.

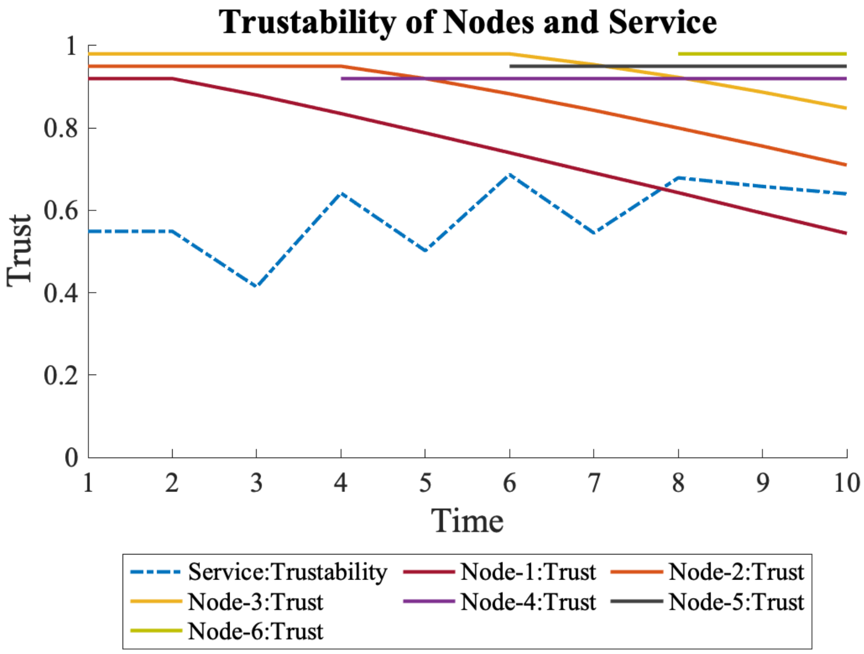

Figure 15. The entire trustability of the system increases each time a new node is launched and connected in parallel to the initial nodes.

Since the continuous decline in trust of the initial nodes keeps decreasing the overall trustability, deactivating those nodes can be considered. Deactivating a parallel connected node with a nonzero trust would also cause a decrease in trustability; however, it is worth exploring the degree of decline resulting from the cost of the nodes.

Figure 16 represents each node's overall trustability and trust when a node is deactivated after a decline in trust. Compared to Figure 15, where the initial nodes are not deactivated, the overall trustability is lower. However, the advantage appears when the cost of the service is considered.

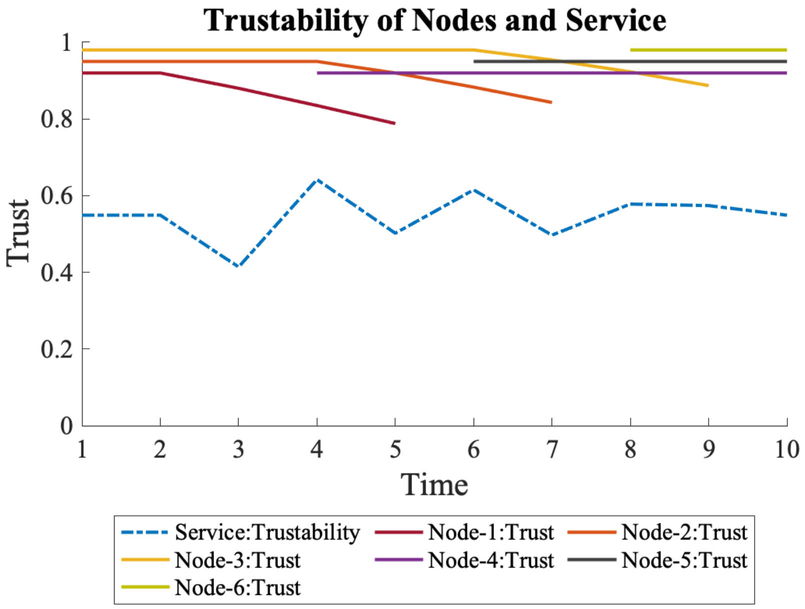

Figure 16. Deactivating a node that is losing trust is one way to keep costs low with the least sacrifice in the overall trustability of the service.

The net utility of the system was calculated using Equation (10). The gain of a running service was assumed to be 500 units, the loss of a down system was considered to be

100 units, and the cost of an active node was assumed to be 50 units. Figure 17 shows the trustability of services running on Cloud-1 and Cloud-2 and their net utilities. Cloud-1 has higher trustability starting at time 4, when the nodes' deactivation started in Cloud-2. The final trustability of Cloud-1 and Cloud-2 are 0.64 and 0.55, respectively, showing a 14% decrease. However, looking at the net utility, which also considers trustability, Cloud-1 has a negative net utility of -16, whereas Cloud-2 has 79.

The result of the net utility function is highly dependent on the choice of gain, loss, and node cost values. In this paper, we identified the importance of employing a trust management framework to actively observe the trustability of the service and each node in order to take the necessary actions that would proactively keep the service alive.

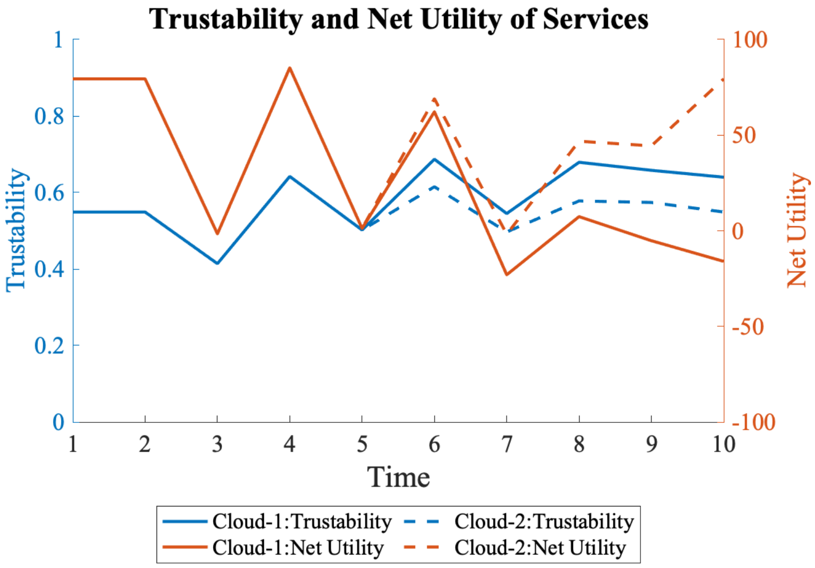

Figure 17. Trustability and net utility of Cloud-1 and Cloud-2. Cloud-1 has higher trustability but lower net utility due to more active nodes.

4. Conclusions

This paper explored the external and internal attacks and incidents occurring in systems with connected sensors and a service deployed on multiple nodes on a cloud. First, two systems consisting of a decision maker and sensors were compared, where the second system had an extra sensor for redundancy. The proposed trust management framework successfully captured the trustability of the sensors and the entire system in both scenarios. An alternative scenario was explored wherein the extra sensor could be activated when necessary, and the results were compared with the prior scenarios.

Furthermore, a sample cloud was simulated with three nodes, where the trust of a node decreased due to some internal incident. The trustability metric captured the overall trustability decline for different incident types. Then, the cloud was updated to add and remove nodes on demand. Each task was supported by an extra node when the individual trust declined in order to keep the overall trustability of the service high. Finally, the cloud systems with and without node deactivation were compared in terms of their trustability measurements and the net utility of the comprehensive service.

The results showed that the utilization of our trust management framework helps in deciding how to mitigate severe consequences of external attacks on sensors or internal incidents on a cloud while considering the net utility of the system. One of the difficulties in this work could be the measurement of the trust of the system parts, such as a sensor or a node, which requires field knowledge for different scenarios; however, it is out of the scope of this paper.

This work can be further extended by considering more realistic and complex scenarios, which include multi-level hierarchies of sensors or nodes. Moreover, a decision maker may rely on other decision makers in addition to sensors. Similarly, a service may depend on other services in addition to its tasks. These scenarios may require the trustability metric formula to be adjusted to specific situations. Future work could also include the cost of trustability measurement operations in the utility function since the historic computation of trust could surge as the number of individual measurements and components increases. Consequently, shifting the defense mechanism from classical, perimeter-based approaches to system resilience is considered to be a long-term objective.

Author Contributions: Conceptualization, S.U., D.K., M.D. and A.D.; methodology, S.U., D.K. and A.D.; software, S.U.; validation, S.U., D.K. and A.D.; writing—original draft preparation, S.U.; writing—review and editing, D.K., M.D. and A.D.; visualization, S.U.; supervision, A.D.; project administration, A.D.; funding acquisition, A.D. All authors have read and agreed to the published version of the manuscript.

Funding: This workwas partially supported by the National Science Foundation under Grant No. 1547411 and by the U.S. Department of Agriculture (USDA), National Institute of Food and Agriculture (NIFA) (Award Number 2017-67003-26057) via an inter-agency partnership between USDA-NIFA and the National Science Foundation (NSF) on the research program Innovations at the Nexus of Food, Energy, and Water Systems.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

Abbreviations

The following abbreviations are used in this manuscript:

5G	Fifth Generation
6G	Sixth Generation
CISA	Cybersecurity and Infrastructure Security Agency
IoT	Internet of Things
MEC	Multi-access Edge Computing
AI	Artificial Intelligence
DM	Decision Maker
S	Sensor
N	Node
UAV	Unmanned Aerial Vehicle
US	United States
USDA	United States Department of Agriculture
NIFA	National Institute of Food and Agriculture
NSF	National Science Foundation

References

1. Armbrust, M.; Fox, A.; Griffith, R.; Joseph, A.D.; Katz, R.; Konwinski, A.; Lee, G.; Patterson, D.; Rabkin, A.; Stoica, I.; et al. A view of cloud computing. *Commun. ACM* **2010**, *53*, 50–58. [CrossRef]
2. Dinh, H.T.; Lee, C.; Niyato, D.; Wang, P. A survey of mobile cloud computing: Architecture, applications, and approaches. *Wirel. Commun. Mob. Comput.* **2013**, *13*, 1587–1611. [CrossRef]
3. Krutz, R.L.; Vines, R.D. *Cloud Security: A Comprehensive Guide to Secure Cloud Computing*; Wiley Publishing: Hoboken, NJ, USA, 2010.
4. Rong, C.; Nguyen, S.T.; Jaatun, M.G. Beyond lightning: A survey on security challenges in cloud computing. *Comput. Electr. Eng.* **2013**, *39*, 47–54. [CrossRef]
5. Pearson, S. Privacy, security and trust in cloud computing. In *Privacy and Security for Cloud Computing*; Springer: London, UK, 2013; pp. 3–42.
6. Xia, F.; Yang, L.T.; Wang, L.; Vinel, A. Internet of things. *Int. J. Commun. Syst.* **2012**, *25*, 1101. [CrossRef]
7. Chettri, L.; Bera, R. A comprehensive survey on Internet of Things (IoT) toward 5G wireless systems. *IEEE Internet Things J.* **2019**, *7*, 16–32. [CrossRef]

8. Goyal, M.; Prakash, S.; Xie, W.; Bashir, Y.; Hosseini, H.; Durresi, A. Evaluating the Impact of Signal to Noise Ratio on IEEE 802.15.4 PHY-Level Packet Loss Rate. In Proceedings of the 2010 13th International Conference on Network-Based Information Systems, Takayama, Japan, 14–16 September 2010; pp. 279–284. [CrossRef]
9. Yang, T.; Ikeda, M.; Mino, G.; Barolli, L.; Durresi, A.; Xhafa, F. Performance Evaluation of Wireless Sensor Networks for Mobile Sink Considering Consumed Energy Metric. In Proceedings of the 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, Perth, WA, Australia, 20–23 April 2010; pp. 245–250. [CrossRef]
10. Xie, W.; Goyal, M.; Hosseini, H.; Martocci, J.; Bashir, Y.; Baccelli, E.; Durresi, A. A Performance Analysis of Point-to-Point Routing along a Directed Acyclic Graph in Low Power and Lossy Networks. In Proceedings of the 2010 13th International Conference on Network-Based Information Systems, Takayama, Japan, 14–16 September 2010; pp. 111–116. [CrossRef]
11. Durresi, A.; Paruchuri, V. Geometric broadcast protocol for sensor and actor networks. In Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA'05), Taipei, Taiwan, 28–30 March 2005; Volume 1, pp. 343–348. [CrossRef]
12. Ullo, S.L.; Sinha, G.R. Advances in smart environment monitoring systems using IoT and sensors. *Sensors* **2020**, *20*, 3113. [CrossRef]
13. Park, S.; Choi, J.Y. Hierarchical anomaly detection model for in-vehicle networks using machine learning algorithms. *Sensors* **2020**, *20*, 3934. [CrossRef]
14. Islam, S.R.; Kwak, D.; Kabir, M.H.; Hossain, M.; Kwak, K.S. The internet of things for health care: A comprehensive survey. *IEEE Access* **2015**, *3*, 678–708. [CrossRef]
15. Hussein, N.H.; Khalid, A. A survey of cloud computing security challenges and solutions. *Int. J. Comput. Sci. Inf. Secur.* **2016**, *14*, 52.
16. Sarwar, A.; Khan, M.N. A review of trust aspects in cloud computing security. *Int. J. Cloud Comput. Serv. Sci.* **2013**, *2*, 116. [CrossRef]
17. Barlet, G. CISA's Strategic Plan Is Ushering in a New Cybersecurity Era. 2022. Available online: https://www.darkreading.com/vulnerabilities-threats/cisa-s-strategic-plan-is-ushering-in-a-new-cybersecurity-era (accessed on 5 December 2022).
18. Easterly, J. CISA Strategic Plan 2023–2025. 2022. Available online: https://www.cisa.gov/sites/default/files/publications/StrategicPlan_20220912-V2_508c.pdf (accessed on 5 December 2022).
19. Li, N.; Mitchell, J.C.; Winsborough, W.H. Design of a role-based trust-management framework. In Proceedings of the 2002 IEEE Symposium on Security and Privacy, Oakland, CA, USA, 12–15 May 2002; pp. 114–130.
20. Zhang, P.; Durresi, A. Trust management framework for social networks. In Proceedings of the 2012 IEEE International Conference on Communications (ICC), Ottawa, ON, Canada, 10–15 June 2012; pp. 1042–1047.
21. Ruan, Y.; Alfantoukh, L.; Durresi, A. Exploring stock market using twitter trust network. In Proceedings of the Advanced Information Networking and Applications (AINA), 2015 IEEE 29th International Conference, Gwangiu, Republic of Korea, 24–27 March 2015; pp. 428–433.
22. Zhang, P.; Durresi, A.; Barolli, L. Survey of trust management on various networks. In Proceedings of the Complex, Intelligent and Software Intensive Systems (CISIS), 2011 International Conference, Seoul, Republic of Korea, 30 June–2 July 2011; pp. 219–226.
23. Ruan, Y.; Durresi, A. A survey of trust management systems for online social communities—Trust modeling, trust inference and attacks. *Knowl.-Based Syst.* **2016**, *106*, 150–163. [CrossRef]
24. Ruan, Y.; Zhang, P.; Alfantoukh, L.; Durresi, A. Measurement theory-based trust management framework for online social communities. *Acm Trans. Internet Technol.* **2017**, *17*, 16. [CrossRef]
25. Ruan, Y.; Durresi, A.; Alfantoukh, L. Using Twitter trust network for stock market analysis. *Knowl.-Based Syst.* **2018**, *145*, 207–218. [CrossRef]
26. Uslu, S.; Kaur, D.; Rivera, S.J.; Durresi, A.; Babbar-Sebens, M. Trust-Based Game-Theoretical Decision Making for Food-Energy-Water Management. In Proceedings of the International Conference on Broadband and Wireless Computing, Communication and Applications, Antwerp, Belgium, 7–9 November 2019; Springer: Cham, Switzerland, 2019; pp. 125–136.
27. Uslu, S.; Kaur, D.; Rivera, S.J.; Durresi, A.; Babbar-Sebens, M. Trust-Based Decision Making for Food-Energy-Water Actors. In Proceedings of the International Conference on Advanced Information Networking and Applications, Caserta, Italy, 15–17 April 2020; Springer: Cham, Switzerland, 2020; pp. 591–602.
28. Uslu, S.; Kaur, D.; Rivera, S.J.; Durresi, A.; Babbar-Sebens, M.; Tilt, J.H. Control Theoretical Modeling of Trust-Based Decision Making in Food-Energy-Water Management. In Proceedings of the International Conference on Broadband and Wireless Computing, Communication and Applications, Yonago, Japan, 28–30 October 2020; Springer: Cham, Switzerland, 2020; pp. 97–107.
29. Kaur, D.; Uslu, S.; Durresi, A.; Mohler, G.; Carter, J.G. Trust-Based Human-Machine Collaboration Mechanism for Predicting Crimes. In Proceedings of the International Conference on Advanced Information Networking and Applications, Caserta, Italy, 15–17 April 2020; Springer: Cham, Switzerland, 2020; pp. 603–616.
30. Kaur, D.; Uslu, S.; Durresi, A. Trust-Based Security Mechanism for Detecting Clusters of Fake Users in Social Networks. In Workshops of the International Conference on Advanced Information Networking and Applications, Matsue, Japan, 27–29 March 2019; Springer: Cham, Switzerland, 2019; pp. 641–650.
31. Rittichier, K.J.; Kaur, D.; Uslu, S.; Durresi, A. A Trust-Based Tool for Detecting Potentially Damaging Users in Social Networks. In *International Conference on Network-Based Information Systems*; Springer: Cham, Switzerland, 2021; pp. 94–104.
32. Li, J.; Li, R.; Kato, J. Future trust management framework for mobile ad hoc networks. *IEEE Commun. Mag.* **2008**, *46*, 108–114.

33. Movahedi, Z.; Hosseini, Z.; Bayan, F.; Pujolle, G. Trust-distortion resistant trust management frameworks on mobile ad hoc networks: A survey. *IEEE Commun. Surv. Tutorials* **2015**, *18*, 1287–1309. [CrossRef]
34. Ruan, Y.; Durresi, A.; Alfantoukh, L. Trust management framework for internet of things. In Proceedings of the Advanced Information Networking and Applications (AINA), 2016 IEEE 30th International Conference, Crans-Montana, Switzerland, 23–25 March 2016; pp. 1013–1019.
35. Ruan, Y.; Durresi, A.; Uslu, S. Trust Assessment for Internet of Things in Multi-access Edge Computing. In Proceedings of the 2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA), Krakow, Poland, 16–18 May 2018; pp. 1155–1161.
36. Ruan, Y.; Durresi, A. A trust management framework for cloud computing platforms. In Proceedings of the Advanced Information Networking and Applications (AINA), 2017 IEEE 31st International Conference, Taipei, Taiwan, 27–29 March 2017; pp. 1146–1153.
37. Kaur, D.; Uslu, S.; Durresi, M.; Durresi, A. A geo-location and trust-based framework with community detection algorithms to filter attackers in 5G social networks. *Wirel. Netw.* **2022**, 1–9. [CrossRef]
38. Pessach, D.; Shmueli, E. A Review on Fairness in Machine Learning. *ACM Comput. Surv.* **2022**, *55*, 51. [CrossRef]
39. Kaur, D.; Uslu, S.; Durresi, A. Requirements for Trustworthy Artificial Intelligence–A Review. In Proceedings of the International Conference on Network-Based Information Systems, Victoria, BC, Canada, 31 August–2 September 2020; Springer: Cham, Switzerland, 2020; pp. 105–115.
40. Thiebes, S.; Lins, S.; Sunyaev, A. Trustworthy artificial intelligence. *Electron. Mark.* **2021**, *31*, 447–464. [CrossRef]
41. Kaur, D.; Uslu, S.; Rittichier, K.J.; Durresi, A. Trustworthy Artificial Intelligence: A Review. *ACM Comput. Surv.* **2022**, *55*, 39. [CrossRef]
42. Varshney, K.R. Trustworthy machine learning and artificial intelligence. *Xrds Crossroads ACM Mag. Stud.* **2019**, *25*, 26–29. [CrossRef]
43. Kaur, D.; Uslu, S.; Durresi, A. Trustworthy AI Explanations as an Interface in Medical Diagnostic Systems. In Proceedings of the International Conference on Network-Based Information Systems, Sanda-Shi, Japan, 7–9 September 2022; Springer: Cham, Switzerland, 2022; pp. 119–130.
44. Uslu, S.; Kaur, D.; Rivera, S.J.; Durresi, A.; Babbar-Sebens, M.; Tilt, J.H. A Trustworthy Human–Machine framework for collective decision making in Food–Energy–Water management: The role of trust sensitivity. *Knowl.-Based Syst.* **2021**, *213*, 106683. [CrossRef]
45. European Comission. Ethics Guidelines for Trustworthy AI; Technical Report. 2018. Available online: https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai (accessed on 5 December 2022).
46. *ISO/IEC TR 24028:2020*; Information Technology—Artificial Intelligence—Overview of Trustworthiness in Artificial Intelligence. International Organization for Standardization: Geneva, Switzerland, 2020.
47. Cao, H.; Zou, W.; Wang, Y.; Song, T.; Liu, M. Emerging Threats in Deep Learning-Based Autonomous Driving: A Comprehensive Survey. *arXiv* **2022**, arXiv:2210.11237.
48. Khan, M.A.; Kumar, N.; Mohsan, S.A.H.; Khan, W.U.; Nasralla, M.M.; Alsharif, M.H.; Żywiołek, J.; Ullah, I. Swarm of UAVs for Network Management in 6G: A Technical Review. *IEEE Trans. Netw. Serv. Manag.* 2022, *early access*. [CrossRef]
49. Saraswat, D.; Verma, A.; Bhattacharya, P.; Tanwar, S.; Sharma, G.; Bokoro, P.N.; Sharma, R. Blockchain-Based Federated Learning in UAVs Beyond 5G Networks: A Solution Taxonomy and Future Directions. *IEEE Access* **2022**, *10*, 33154–33182. [CrossRef]
50. Velliangiri, S.; Manoharan, R.; Ramachandran, S.; Rajasekar, V. Blockchain Based Privacy Preserving Framework for Emerging 6G Wireless Communications. *IEEE Trans. Ind. Inform.* **2022**, *18*, 4868–4874. [CrossRef]
51. Uslu, S.; Kaur, D.; Rivera, S.J.; Durresi, A.; Durresi, M.; Babbar-Sebens, M. Trustworthy Acceptance: A New Metric for Trustworthy Artificial Intelligence Used in Decision Making in Food-Energy-Water Sectors. In Proceedings of the International Conference on Advanced Information Networking and Applications, Toronto, ON, Canada, 12–14 May 2021; Springer: Cham, Swizerland, 2021; pp. 208–219.
52. Uslu, S.; Kaur, D.; Rivera, S.J.; Durresi, A.; Durresi, M.; Babbar-Sebens, M. Trustworthy Fairness Metric Applied to AI-Based Decisions in Food-Energy-Water. In Proceedings of the International Conference on Advanced Information Networking and Applications, Sydney, NSW, Australia, 13–15 April 2022; Springer: Cham, Switzerland, 2022; pp. 433–445.
53. Kaur, D.; Uslu, S.; Durresi, A.; Badve, S.; Dundar, M. Trustworthy Explainability Acceptance: A New Metric to Measure the Trustworthiness of Interpretable AI Medical Diagnostic Systems. In Proceedings of the Conference on Complex, Intelligent, and Software Intensive Systems, Asan, Republic of Korea, 1–3 July 2021; Springer: Cham, Switzerland, 2021; pp. 35–46.
54. Ruan, Y.; Durresi, A. A trust management framework for clouds. *Comput. Commun.* **2019**, *144*, 124–131. [CrossRef]