Journal of
Sensor and Actuator Networks

MDPI

Article

A Trust-Influenced Smart Grid: A Survey and a Proposal

Kwasi Boakye-Boateng [1,*], Ali A. Ghorbani [1] and Arash Habibi Lashkari [2]

[1] Canadian Institute for Cybersecurity (CIC), Faculty of Computer Science, University of New Brunswick (UNB), Fredericton, NB E3B 5A3, Canada; ghorbani@unb.ca
[2] School of Information Technology, York University, Toronto, ON M3J 1P3, Canada; ahabibil@yorku.ca
* Correspondence: kwasi.boakye-boateng@unb.ca

Abstract: A compromised Smart Grid, or its components, can have cascading effects that can affect lives. This has led to numerous cybersecurity-centric studies focusing on the Smart Grid in research areas such as encryption, intrusion detection and prevention, privacy and trust. Even though trust is an essential component of cybersecurity research; it has not received considerable attention compared to the other areas within the context of Smart Grid. As of the time of this study, we observed that there has neither been a study assessing trust within the Smart Grid nor were there trust models that could detect malicious attacks within the substation. With these two gaps as our objectives, we began by presenting a mathematical formalization of trust within the context of Smart Grid devices. We then categorized the existing trust-based literature within the Smart Grid under the NIST conceptual domains and priority areas, multi-agent systems and the derived trust formalization. We then proposed a novel substation-based trust model and implemented a Modbus variation to detect final-phase attacks. The variation was tested against two publicly available Modbus datasets (EPM and ATENA H2020) under three kinds of tests, namely external, internal, and internal with IP-MAC blocking. The first test assumes that external substation adversaries remain so and the second test assumes all adversaries within the substation. The third test assumes the second test but blacklists any device that sends malicious requests. The tests were performed from a Modbus server's point of view and a Modbus client's point of view. Aside from detecting the attacks within the dataset, our model also revealed the behaviour of the attack datasets and their influence on the trust model components. Being able to detect all labelled attacks in one of the datasets also increased our confidence in the model in the detection of attacks in the other dataset. We also believe that variations of the model can be created for other OT-based protocols as well as extended to other critical infrastructures.

Keywords: cybersecurity; trust; Modbus; Modbus TCP; fieldbus; substation; substation security; risk; Smart Grid; advanced persistent threats

Citation: Boakye-Boateng, K.; Ghorbani, A.A.; Lashkari, A.H. A Trust-Influenced Smart Grid: A Survey and a Proposal. *J. Sens. Actuator Netw.* **2022**, *11*, 34. https://doi.org/10.3390/jsan11030034

Academic Editor: Lei Shu

Received: 28 May 2022
Accepted: 5 July 2022
Published: 11 July 2022

1. Introduction

The Smart Grid is the transformation of the traditional grid which can be combined with cyber devices to automate monitoring and control as well as include a two-way communication between systems [1]. The Smart Grid's performance, just like that of the traditional grid, is centred on factors such as distribution, transmission, and generation. The coupling of the traditional power grid's physical components and the cyber infrastructure has made the creation and continuous improvement of the Smart Grid possible. The diverse nature of the Smart Grid introduces varying applications and the integration of components such as electric vehicles, renewable energy resources, and variants of distributed power generators. Smart Grid has also introduced and improved vendor-independent standards that devices must conform to, thus allowing the seamless operation and integration of these devices into the Smart Grid.

Unfortunately, the cyber infrastructure's integration into the power grid increases the attack vector of the Smart Grid, thereby making the security of the Smart Grid of

paramount importance. In response, research has been undertaken under varying topics such as encryption [2], generation and management of cryptographic keys [3], privacy [4], risk assessment [5], and trust. Trust within the Smart Grid is important for determining whether an action, transaction, or communication is malicious or not. In the case of the notorious Stuxnet [6], there is the possibility that trust could have been implemented in devices to ascertain the legitimacy of malicious commands before responses or actions are taken on those commands.

We also observed that research on trust has not received the considerable attention that it deserves within the Smart Grid even though it abounds in other research areas such as E-commerce. Furthermore, other branches of security within the Smart Grid have largely received more contributions than trust. As of the time of writing this paper, no study assessing trust within the Smart Grid exists. As of the time of this study, we noticed that there were limited trust models that detected operational faults within the substation. However, these models could neither determine whether the faults were malicious or not, nor detect obvious or stealthy malicious attacks within the substation. Such attacks were predominant within advanced persistent threats (APTs).

The contributions of this study are as follows:

- We present a mathematical formalization of trust within the context of Smart Grid devices.
- We categorize the existing trust-based literature within the Smart Grid under the NIST conceptual domains and priority areas, multi-agent systems, and the derived trust formalization.
- We present a proposed novel substation-based trust model and implement a Modbus variation to detect final-phase attacks. We believe other protocol variants of the trust model can be created and developing these will be addressed in future work.
- The variation is tested against two publicly available Modbus datasets (EPM and ATENA H2020) under three kinds of tests, namely external, internal, and internal with IP-MAC blocking.
- The tests were performed from a Modbus server's point of view and a Modbus client's point of view.
- All attacks were detected and the behaviour of attacks was revealed based on their impact on the trust model's components.

In this paper, we provide a background on the priority areas and conceptual domains of the Smart Grid as described by the National Institute of Standards and Technology (NIST) in Sections 2 and 3. We then present a background on trust, its definitions, and trust-related attacks in Section 4. We categorize existing literature from Sections 5–8. We present a proposed trust model and its Modbus variation from Sections 9–11. Implementation and results are presented in Sections 12 and 13, respectively. We provide our conclusions and future work in Sections 14 and 15. We also included a table of notations in Appendix A to be used as reference for the equations in the paper.

2. NIST Priority Areas On Smart Grid

The inclusion of a cyber infrastructure introduced a deficiency of myriad standards, which made maintaining the efficiency of the Smart Grid extremely challenging. In light of that, NIST identified nine key priority areas to be focused on to tackle these challenges [7]. These areas are discussed in this section.

2.1. Energy Storage

One major challenge in the power industry is the storage of energy. Because of the immense difficulties posed by such storage, supply and demand are carefully balanced. This challenge brings about the need to invest and investigate new technologies to store energy, which will improve the efficiency within the grid from supplier to consumer.

2.2. Wide-Area Situational Awareness (WASA)

Monitoring various components within the Smart Grid is salient to ensure their optimization. This guarantees that processes of demand and supply, as well as utilization forecasts, are facilitated. Thus, novel technologies and strategies are required to create tools that monitor and display these components within the Smart Grid.

2.3. Advanced Metering Infrastructure (AMI)

Power usage by consumers is a key parameter in observing demand within the Smart Grid. In the traditional grid, meters were manually read and recorded before being computed to know the actual utilization within a given period. The introduction of the Smart Grid assures the near-real-time monitoring of power usage with AMI. AMI creates a dual-channel network between the smart meters and business systems of utility providers. This enables the collection and distribution of meaningful data to customers and utility providers as well as competitive retail suppliers. Such information can be used to implement residential demand responses. Even though there are many different designs of AMI, it consists of communications software and hardware and their associated system and data management software.

2.4. Distributed Energy Resources (DERs)

DERs are resources that generate and/or store electricity for a local distribution system or a facility within that system. As such, DERs connect to these systems. DERs include combined heat and power (CHP) generators, electric vehicles/plug-in electric vehicles (PEVs), battery storage systems, solar panels, microgrids, and battery storage systems [8,9]. Because these technologies are relatively new, they continuously evolve. One key concern is using these resources to ensure a resilient, safe, and uninterrupted power grid and safeguarding the efficient generation, utilization, and storage of power from these resources.

2.5. Distribution Grid Management

Distribution grid management systems integrate customer operations, networked distribution systems, and transmission systems with actual physical components, such as transformers, feeders, circuit-breakers and relays, to enable real-time functionalities such as the monitoring of system performances and load utilization [7]. Thus, the automation of distribution systems is important to operations of the Smart Grid, especially where systems such as AMI and PEVs are deployed to provide benefits such as reductions in peak loads, providing field engineers with malfunctioning devices' locations, and increased reliability.

2.6. Network Communications

Communication within the Smart Grid is important to ensure real-time monitoring, operations, and maintenance within the Smart Grid. Therefore, various technologies such as fibre-optics, wireless, and cellular (currently trending is 5G) are required in strategic areas or locations to aid in Smart Grid operations. Different routing algorithms are also required to ensure fast communication for the time-sensitive operations of some devices within the Smart Grid. Access to public and private communication networks will be required with various restrictions in place. Furthermore, critically important is ensuring that there is no collision or loss of messages during their transmission. Power network interfaces are required for long-distance transmission, and cost-effective solutions are always required. The efficient translation of protocols is also required as well as global standards to ensure that vendors can comply, thereby making communication seamless.

2.7. Demand Response and Consumer Energy Efficiency

Technologies to balance supply and demand are being used by electricity suppliers and system planners. These technologies allow them to provide incentives (mostly financial) and mechanisms for consumers that lead to the efficient use of power during unstable power

periods or peak periods. By providing detailed information to clients about consumption, they can save energy by engaging in practices and investing in devices that ensure the efficient utilization of power. Offering time-based rates such as critical peak rebates, variable peak pricing and time-of-use pricing can allow customers to take part in demand response efforts. Customers could allow utility companies to use direct load control programs to cycle water heaters and air conditioners on and off during peak periods in exchange for lower bill charges or incentives that may be financial or non-financial.

2.8. Electric Transportation

Clean energy ensures reduced carbon emissions, reduced dependency on fossil fuel to drive the economy, and reduced carbon footprint for nations. Thus, the large-scale usage and patronage of PEVs are essential in ensuring that this happens. Technologies to ensure the cost-effective mass creation of these electric vehicles and their storage capacity are crucial to ensure that this happens.

2.9. Cybersecurity

In a world where everything is being relocated to the cyber-domain, cybersecurity is critical to ensure the safety, availability, and reliability of the Smart Grid. It is very important to ensure that the operations of the Smart Grid are not adversely affected when security is applied within the grid. Cybersecurity plays a critical role in the operations of previously mentioned areas (Figure 1). There has been research into (but not limited to) network communication [10,11], demand response [12,13], PEVs [14,15], AMI [16,17] and DER [18,19]. This research includes encryption [19], privacy [20], intrusion detection and prevention [21], and trust. In this paper, we present a survey on the research on trust within the Smart Grid, especially within the priority areas and conceptual domains of the Grid. In terms of systems and trust, it is required that systems be cognitive to be able to trust each other. It is for this reason that we also investigate the application of trust in multi-agent systems' research within the Smart Grid. We also propose a trust model for substations within the Smart Grid.

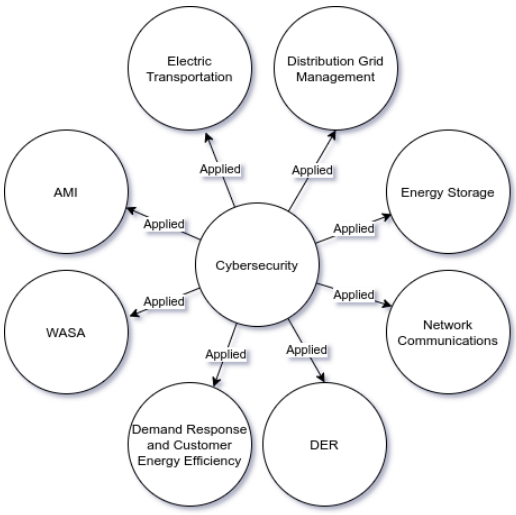

Figure 1. NIST priority areas: Importance of cybersecurity in priority areas.

3. NIST Conceptual Domain Model

The conceptual domain model represents seven logical domains within the Smart Grid [7]. These domains represent the present and near-future view of the Smart Grid (Figure 2). The domains communicate with each other through interfaces. Figure 3 shows the mapping of legacy systems in the grid to the conceptual domains.

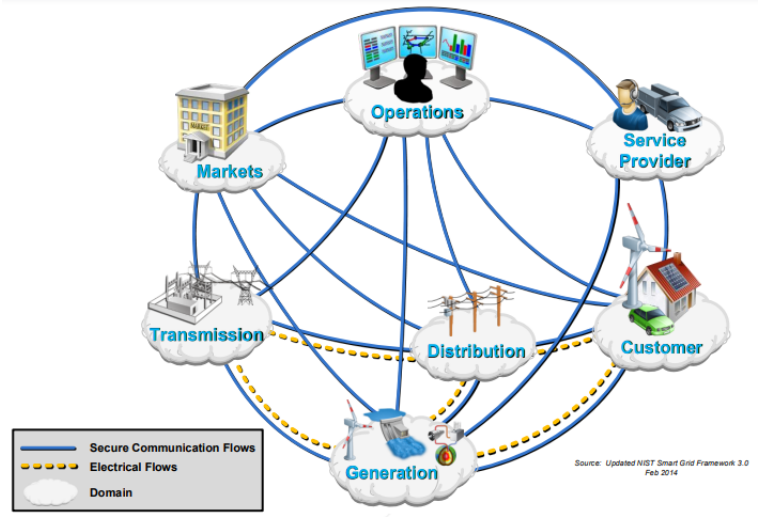

Figure 2. NIST conceptual domains [7].

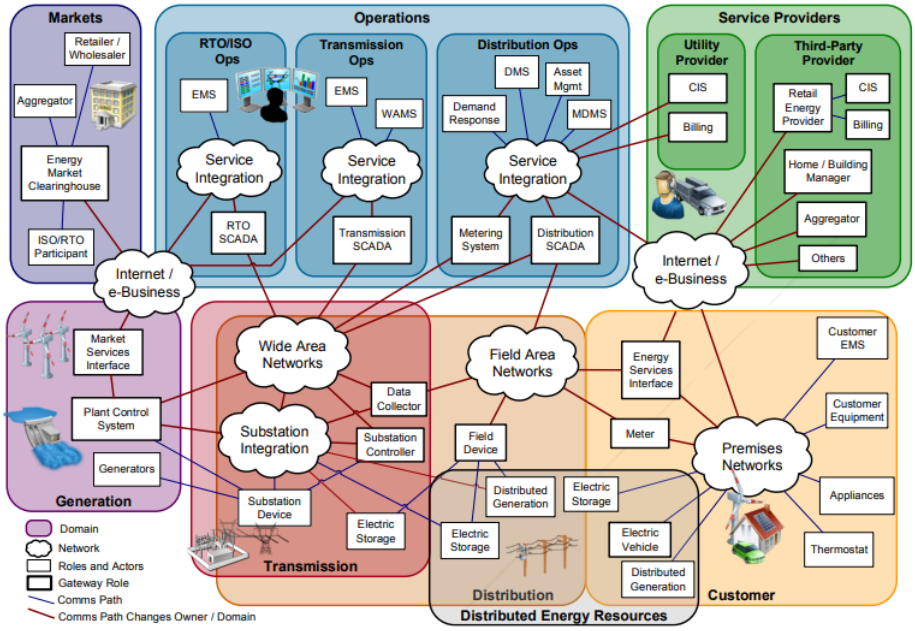

Figure 3. Mapping of legacy systems to conceptual domain [7].

3.1. Generation Domain

This is the domain where power or electricity is generated from renewable or non-renewable forms of energy, and applications in this domain are the first processes when it comes to the delivery of power to customers [22]. It is from here that power is transferred to the transmission or the distribution domain. Thus, the connections with those two domains must remain reliable because power cannot be served to customers without it. Applications that can be found in this domain are asset management, protection, measurement, records/logging and control.

3.2. Transmission Domain

The transmission domain is responsible for the bulk transfer of electrical power to the distribution domain from the generation domain through the use of multiple substations. A transmission network is usually managed and operated by a transmission-owning entity with the primary responsibility to ensure stability on the electrical grid by balancing supply (power generation) with demand (power consumption) across the transmission network. A Supervisory Control and Data Acquisition (SCADA) system, which comprises

a communication network, control devices and field monitoring devices, is used to monitor the transmission network.

3.3. Distribution Domain

The distribution domain is electrically connected between the transmission domain and the customer domain. The electrical distribution system may be structured in a varied number of ways such as meshed, looped or radial—and each structure affects the reliability of the system. Initially, the communications interfaces within this domain were unidirectional and hierarchical, but now they work in a bi-directional manner. Typical applications within this domain are measurement and control, substation, DERs, distribution generation and storage.

3.4. Operations Domain

This domain ensures that the power system runs smoothly. A regulated utility is assigned the responsibility of ensuring this. Even though some of the functions in this domain may be provided by the service provider as the Smart Grid continuously evolves, there will always be core functions maintained in this domain. Typical applications in this domain are customer support, fault management, operation planning, monitoring, network calculations, maintenance and construction, analysis, records and assets, control, extension planning and reporting, and statistics.

3.5. Service Provider Domain

The service provider domain provides support to other domains such as home energy generation, the management of energy use, and billing and customer account management. Its communication with the operations and markets domain is critical for situational awareness, system control and enabling economic growth. Typical applications in the service provider domain include building management, customer management, installation and management, account management, billing and building management.

3.6. Markets Domain

The sale and purchase of grid assets are conducted in the Markets domain, hence its importance to ensure that communications within this domain are transparent and reliable. There is the balance of supply and demand as well as the exchange price within the power system that is ensured by this domain. It must also be noted that due to the evolving nature of the Smart Grid, the market domain is bound to evolve, which in turn will define the Smart Grid in the future. The market domain communicates with the entity that controls the assets (operations domain), the customer domain and the other domains that supply the assets. The efficient matching of demand for power with the consumption of power is dependent on the domain of the market; thus, the communication flow between that domain and the domains that supply the power is critical. Bulk generation and DERs (which are usually served through aggregators) are examples of power suppliers, with DER more likely to become greater partakers as the interactive nature of the grid increases. Typical applications in the market domain include market management, DER aggregation, market operations, trading, ancillary operations and retailing.

3.7. Customer Domain

The customer is the main beneficiary of the Smart Grid and is the reason the Grid was created. The sole purpose of the customer is to consume the electricity generated by the grid. The customer domain is usually divided into home, commercial/building and industrial domains due to the difference in their energy demands. Each sub-domain has a meter and an interface that connects to other domains for utility-to-customer interactions. This may be done over the Internet or the AMI. Home or building automation is one of the applications in the customer domain that relies on these interfaces to function. Home automation allows the control of appliances within the house. Industrial automation,

which is similar to home automation, also allows the control of industrial processes such as manufacturing. The interfaces also allow the storage of energy in thermal energy units and batteries as well as the generation of energy from renewable sources such as solar panels that are close to the customer. Although the customer domain communicates with and is electrically connected to the distribution and generation domains, it communicates with the service provider, operations, and market domains.

4. Trust

The world would not function without trust. Without trust, it would be difficult for interactions and/or transactions to exist. As a concept, trust is fundamental in the building and maintenance of stability in human relations. Trusting someone or something helps create interactions between people and organizations. In the digital age, with the current existence of virtual markets and communities, the interest in trust has matured and as such, can be expanded into other domains. Thus, any effort undertaken towards the proper management of trust by sharing information that enables interactions between participants in the open environment is essential and challenging. It is worth noting that trust is only useful in uncertain situations where people or agents must cooperate to achieve goals.

4.1. Trust Definition and Formalization

According to the literature, trust has many definitions. A definition from the social sciences states that trust is the degree of subjective belief about the behaviours of a particular entity [23]. Trust is also defined as an agency's subjective probability of performing a particular act [24]. In this paper, we define the trusting entity as the *agent* and the entity being trusted as the *subject*. Marsh [25] describes three levels of trust, namely basic trust, general trust, and situational trust. Basic trust is the general trusting disposition of an agent. General trust is the trust that an agent has on a subject at a certain time. Situational trust is the trust that the agent has on the subject, taking into account a certain situation.

It must be noted that trust has been applied in different contexts, thus the notion that trust has many definitions. Thus, the design of trust models is required to be within a context or in terms of the system being designed. Thus, the factors being chosen to design the trust model must be on objective grounds to ensure that the trust being modelled is also objective. Hence, the difficulty in modelling trust. Regardless, trust models must have a component that must accept the risk because, without the assessment of risk, there is no trust.

NIST defines risk as: *A measure of the extent to which an entity is threatened by a potential circumstance or event* [26]. Thus, for an agent, a_i, and a subject, a_j, we define the risk, r_{ij}, of a transaction, α_{ij}, involving a_i and a_j as a function as shown in (1). There must also be a component of knowledge, k^t_{ij}, within the trust model. Before and after a transaction, knowledge about α_{ij} and previous transactions (k'_{ij}) with the subject, the environment (k_e), knowledge of a_j, k_{a_j}, and the time period (t), are also of prime importance in determining trust. We formulate knowledge as shown in (2). k'_{ij} is a collection of transactions before the current transaction, and this is formulated in (3).

$$r_{ij} = f(\alpha_{ij}) \tag{1}$$

$$k^t_{ij} = f(\alpha_{ij}, k_e, k'_{ij}, k_{a_j}, t) \tag{2}$$

$$k'_{ij} = \{k^{t-1}_{ij}, k^{t-2}_{ij}, ..., k^0_{ij}\} \tag{3}$$

$$T_{ij} = f(a_i, a_j, r_{ij}, k_{ij}, T^{'}_{ij}) \tag{4}$$

$$T_{ik} \nsim T_{jk} \; where \; 1 - T_{ij} \approx 0, \; 1 - T_{jk} \approx 0 \tag{5}$$

$$T_{ik} \sim T_{jk} \; where \; 1 - T_{ij} \approx 0, \; 1 - T_{jk} \approx 0 \tag{6}$$

Thus, with risk, r, and knowledge, k_{ij}^{t}, the decision on trust can be made. Therefore, trust, T_{ij}, can be expressed as the output of a function that takes a tuple of elements as shown in (4) where $T_{ij}^{'}$ is the previous trust value between a_i and a_j. The $T_{ij}^{'}$ has an influence on the decision for a_i to trust a_j to undertake α_{ij}. Trust is represented as a continuous variable over a specified range usually $-1 \leq T \leq 1$ or $0 \leq T \leq 1$ where 1 represents complete trust, -1 represents complete mistrust and 0 represents no trust. It must be noted that the transitive property of trust may or may not exist. In a situation where it does not exist, for three agents a_i, a_j, and a_k, the fact that a_i trusts a_j and a_j trusts a_k does not mean that a_i trusts a_k (see (5)). In a situation where transitivity exists, it means that a_i trusts a_j and a_j trusts a_k, therefore, a_i trusts a_k (see (6)).

Trust can be directly or indirectly evaluated. Direct trust is calculated based on direct interactions between the agent and the subject. The default definition of trust is direct trust and that is formulated in (4). In the situation where no interaction exists between the agent, a_i, and subject, a_j, trust is built based on opinions from other agents about the subject; this is termed indirect trust. As formalized in (7), in an environment of n agents, trust is computed based on the recommendation of, at most, $n-2$ agents.

$$T_{ij} = f(T_{i_{+1}j}, T_{i_{+2}j}, ..., T_{n_{-2}j},) \tag{7}$$

4.2. Trust-Based Attacks

In ensuring that trust mechanisms do not work in an environment, adversaries employ different attacks or strategies [27,28]. Some of these attacks are as follows:

- *Misleading feedback attack:* In this attack, a compromised agent feeds bad reports or recommendations to other nodes to denigrate agents with good reputations. It is also known as bad-mouthing attack or betrayal attack.
- *Sybil attack:* This attack involves a malicious agent within the system creating fake identities to create a larger influence over other agents using false rankings.
- *Newcomer attack:* This attack involves the malicious agent reintroducing itself as a new agent within the system in an attempt to erase its history of bad scores.
- *Ballot-stuffing attack:* In this attack, malicious agents collude by providing inaccurate recommendations or reports in an attempt to take over the system. It is also known as collusion attack.
- *On–off attack:* This attack involves a malicious agent repeatedly switching between being honest and dishonest in an attempt to be undetected. It is also known as inconsistency attack.

5. Trust: State of the Art in Smart Grid

In this section, we present literature on trust within the Smart Grid, categorized by the priority areas, conceptual domains, and trust definitions—after which we briefly discuss our observations. We searched the IEEE, Science Direct, Scopus, Web of Science, ACM, and Springer Link databases to find literature by using the keywords trust, reputation, trust management, mistrust, and trust model. We further reduced the papers by pairing each keyword with each of the following keywords: cyber-physical systems, critical infrastructure, distributed energy resources, micro-grids, smart grid, smart meters, substations, advanced metering infrastructure, building automation and control systems, distribution automation, and industrial control systems. We streamlined the list by reading the abstracts to ensure that the papers were relevant to the subject matter. The remaining papers were scrutinized and categorized or left out if they were not relevant to the subject matter.

5.1. Research Areas

Cheng et al. sought to detect the credibility of data from different sources by establishing trust from the said sources [29]. Though they were not specific about which part of Smart Grid they were working on, their work implied that it could be used in all areas of the Smart Grid because it deals with big data. In their paper, they used trust and credibility

interchangeably. Even though the knowledge component exists in terms of previous trust values and a forgetting rate, the measure of risk on the data from the data source and the data source itself was not computed. There were no tests against trust-based attacks.

Moving away from big data to secure routing, another paper sought to compute trust for secured routing in wireless-based communications in the Smart Grid [30]. Network-based features such as the average transmission rate, buffering capacity and time-to-live (TTL) are used to compute trust. Their algorithm first computes direct trust between nodes; indirect trust based on recommendations from other nodes; and finally uses that information to compute how to route information from one node to another within the Smart Grid communication infrastructure. This algorithm would work best in AMI but not in the generation and distribution domains of the Smart Grid where communications are more wired than wireless. This paper improved their previous trust model to identify benign and malicious nodes based on various features using a combination of Bayes, Dempster–Schafer and Fuzzy theory [31]. They employed a water cycle algorithm (WCA) to improve its efficiency and tested it using an NS-2 simulator. The parameters used are clear indicators of the knowledge component of trust; however, there was no measurement for risk to show the impact should a node be wrongfully trusted. The algorithm was also not tested against trust-based attacks.

Another paper also proposed a fuzzy logic-based trust model to ensure secure routing in the network [32]. It computes a global trust value by computing direct and indirect trust to allow nodes to make decisions on compromised nodes. They tested their work against trust-based attacks, but their algorithm had no risk component.

Still focusing on routing, Xiang et al. presented a trust-based geographical routing protocol which placed trusted nodes in a trust list [33–35]. To be part of the trusted list, the node was required to have a good performance ratio as well as a good recommendation from other nodes. Based on that list, a routing algorithm is implemented to route from one trusted node to another. Their work did not include a risk component and was not tested against trust-based attacks, even though it was tested against WSN-based DOS attacks. Their experiment was simulated using a Java-based simulator called J-Sim.

Though not creating their trust model, Bello et al. explored the impact of transitivity in network topology in the performance evaluation of the famous EigenTrust model [36]. They demonstrated that a network containing established transitivity connections implied that a benevolent node was quickly identified by a node, thereby reducing the average energy consumption. An improved version was tested against trust-based attacks and showed that structural similarity has an impact on robustness against trust-based attacks and malicious nodes [37].

In trying to detect a compromised node in a network, a trust management model was proposed based on fuzzy logic using the packet error rate, interaction duration and packet loss rate as features [38] to compute trust. There was no risk component in the calculation, and neither was the algorithm tested against trust-based attacks. The trust model was simulated using Xfuzzy-3.5.

Moving away from networks, and still within AMI, Pliatsios et al. computed trust based on three features, namely consumption, polling, and connection to detect malicious devices [39]. The continuous-time Markov chain was utilized to compute the trust value of a node. It was purely tested with numerical parameters. The trust value of a device was decreased or increased in unit steps within the range of 1–3 (inclusive) depending on the behaviour of the device. The state of the Markov chain stores the state of a previous interaction. However, the risk component does not exist to determine the extent of a possible threat on or from the device. Furthermore, an on–off attack can be used to ensure that the device's trust value is maintained.

In tackling meter tampering within the AMI, Pradhan et al. did the reverse of calculating trust by using mistrust [40]. Their algorithm involved comparing the presented data with houses and actual data from smart meters to see whether a house is being truthful or

not. A dishonest house is added to a mistrust table. Their algorithm has no risk component and was not tested against trust-based attacks.

In tackling cascading power failures, a trust management toolkit was proposed, which computes a trust value using the simple trust algorithm [41] which uses the threshold of grid values as input [42]. With the trust values being attained and Djikstra's shortest path algorithm, it allows the flow of power in an optimal direction to prevent cascading failures. This work was improved upon to create a special protection system (SPS) that implemented a trust mechanism that is con-resistant and mitigates transient instabilities (being aperiodic of time) within the grid by using load-shedding strategies [43]. One of the key features in calculating trust values was ensuring that a node reports a frequency value around a specific threshold. There was no risk component, and their work was not tested against trust-based attacks.

Other papers assume that trust is already manifest in firewalls, intrusion detection systems (IDSs) and other security devices and therefore apply the term trust nodes for these devices. Thus, their research involves placing them in vantage points within the AMI [44–48] or SCADA network [49] and computing an optimal routing algorithm for them, especially when a node is compromised. These papers do not include any computations of trust because they assume that trust is already embedded in the devices.

5.2. Discussion

Concerning Table 1, it can be observed that the majority of the papers reviewed focused on AMI and network communications areas. Only one paper [29] fits across all the priority areas. Only two papers [42,43] were specifically focused on distribution grid management. Trust in the research areas of energy storage, electric transportation, demand response and consumer energy efficiency, WASA and DER is lacking.

In Table 2, research by Cheng et al. [29] covers all seven conceptual domains. Only two papers specifically cover transmission, distribution, generation and operation domains. The rest were focused on customer and service provider domains.

In Table 3, none of the papers had a risk component for computing trust, and only two of the papers [32,37] tested their work against trust-based attacks. The knowledge component of most of the papers did not include previous transactions or states; thus, trust was computed based on the values of parameters that were provided for computation. Only two papers [38,39] implemented direct trust, and the rest computed both direct trust and indirect trust.

Table 1. Research on trust in Smart Grid categorized based on NIST priority areas.

NIST Priority Areas							
Distribution Grid Management	**Energy Storage**	**AMI**	**Electric Transportation**	**Network Communications**	**Demand Response and Consumer Energy Efficiency**	**WASA**	**DER**
[29,42,43,49]	[29]	[29–31,39] [33–35,38] [32,36,37,40] [44–47] [48,49]	[29]	[29–31,38] [33–36] [32,37,44,45] [46–49]	[29]	[29]	[29]

Table 2. Research on trust in Smart Grid categorized by NIST conceptual domains.

NIST Conceptual Domains						
Transmission	**Generation**	**Distribution**	**Markets**	**Customer**	**Service Provider**	**Operations**
[29,42,43]	[29,42,43]	[29,42,43]	[29]	[29–31,39] [33–35,38] [32,36,37,40] [44–47] [48]	[29–31,39] [33–35,38] [32,36,37,40] [44–47] [48]	[29,42,43,49]

Table 3. Research on trust in Smart Grid categorized by trust components.

Trust Components				
Direct Trust	**Indirect Trust**	**Tested Against Trust Attacks**	**Risk Component**	**Knowledge Component**
[29–31,39] [33–35,38] [36,37,42,43] [32,40]	[29–31,33] [34,35,42,43] [32,36,37,40]	[32,37]	-	[29–31,39] [36,38,42,43] [32,37,40]

6. Trust: State of the Art in Substations

Substations, aside from other functions, are responsible for transforming low voltage into high voltage or vice versa [50]. They are considered integral to the transmission and distribution of power within the Smart Grid. Substation automation systems (SASs), consisting of the station level, process level, and bay level, enable the integration of substations into the Smart Grid. The station level contains SCADA and some variations of HMI; the bay level comprises IEDs; and the process level comprises high-voltage primary devices (see Figure 4). IEDs are responsible for controlling circuit breakers which are responsible for the connection or disconnection of power lines. It is SCADA that controls the IEDs by sending commands to them.

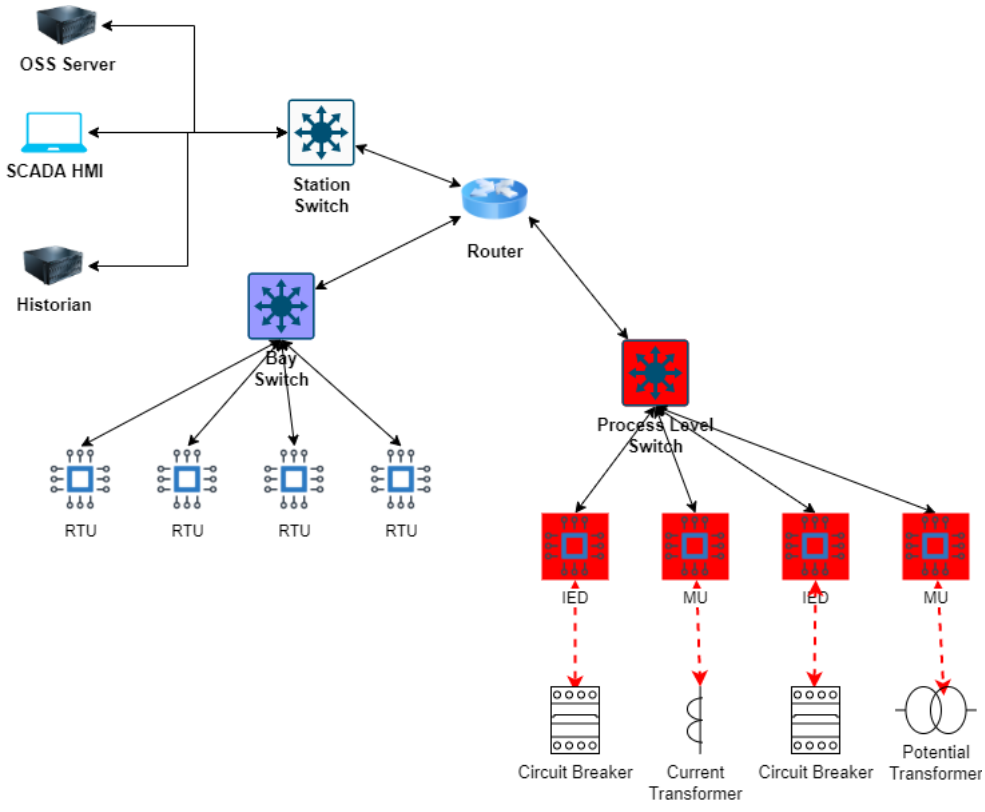

Figure 4. Substation automation system.

6.1. Research Areas

Trust has been stated as an important reflection of the state of the substation, the execution of legitimate commands of devices within the substation and the dissemination of sensitive substation information [51]. To detect malicious nodes in the protection zones of substations, trust was implemented in wireless sensor nodes [52,53] by using their wireless range. It must be noted that most substations that exist, at the time of this paper, do not use wireless sensor nodes in protection zones for substations but rather use IEDs which are serial-based or Ethernet-based.

Another paper presented the measurement of trust between substations by the use of behavioural pattern analysis [54,55]. The analysis used machine learning and statistical tools and used logs from the security gateway of substations as the source of data. These logs contained communication between substations. They computed a threat value to substations based on which the inverse was the trust value. However, the analysis is external to the substations, and therefore, an attack within a substation is likely to be over before an analysis is completed. Furthermore, most attacks originate from SCADA with legitimate commands, and these can go undetected.

Nasr et al. [56] built a system to secure SCADA from deontological threats. The system aims to limit the access of an attacker or a naive/unskilled operator to a critical substation. The performance of an operator in controlling remote substations and resolving alarms is considered in determining the operator's trustworthiness.

Rashid et al. designed a trust system for securing IEC 61850 GOOSE communication [57]. The untested trust system comprised modules that mimicked firewall policies, checked frame formats and access control.

6.2. Discussion

None of the papers tested their work against trust-related attacks nor did they include a risk component in their models (see Table 4). The knowledge component of most of the papers did not include previous transactions or states and as a result, trust was computed based on the values of parameters that were provided for computation. Only two papers implemented both direct and indirect trust. None of the papers tested their work against trust-related attacks.

Table 4. Research on trust in substation categorized by trust components.

Trust Components				
Direct Trust	**Indirect Trust**	**Tested Against Trust Attacks**	**Risk Component**	**Knowledge Component**
[52–55] [56,57]	[52,53]	-	-	[52–55] [56,57]

7. Multi-Agent Systems (MASs)

A multi-agent system (MAS) is a system consisting of two or more intelligent agents [58]. An intelligent agent is described as an entity with four characteristics, namely social ability, reactivity, pro-activeness, and autonomy. Social ability requires that the agent should be able to interact with other agents. This is often mistaken as just the exchange of messages. However, it requires the ability to cooperatively interact and negotiate or in simple terms; agents should be able to converse. Reactivity requires that when there are changes to the environment in which the agent is in, the agent must react promptly; and based on its goals and those changes, the agent must take some appropriate action. Pro-activeness requires that the agent must change its dynamically behaviour to achieve its goals. Autonomy requires that agents must operate without any intervention from humans or any external system.

An MAS has an overall objective or goal to which each agent's goals within the MAS must contribute to the achievement of that overall objective. There are three kinds of MAS architectures, namely centralized, decentralized, and hybrid. Centralized architecture has agents reporting to a central agent from whom the agents await instructions. Decentralized architecture has agents communicating with each other in a clustered manner, with each having the same level of priority. In the case of centralized architecture, the demise of the central agent spells the demise of the MAS. The optimization of MAS goals is challenging with a decentralized architecture because of the local nature of the connection between agents. The hybrid architecture combines the two previous architectures to utilize their advantages.

MAS has been implemented in microgrids [59], demand side management [60], smart meters [61], optimal power flow and energy-sharing [62], and Smart Grid simulation [63].

7.1. MAS Tools

The development of intelligent agents and MASs requires tools to make this feasible. The major software frameworks identified are presented in this section.

7.1.1. JADE

Java Agent Development Framework (JADE) is a software framework fully developed in the Java language [64–66]. JADE uses middleware to simplify the implementation of MAS, which ensures its implementation across a platform-independent distributed system. It also incorporates a set of graphical tools that are essential in remote configuration, debugging, and deployment. JADE is also free to use and is compliant with the specifications of the Foundation for intelligent physical agents (FIPA).

7.1.2. ZEUS

Zeus [65] is an open source agent development platform developed with the Java language. It is FIPA-compliant and supports knowledge query and manipulation language (KQML). It has, however, been discontinued.

7.1.3. VOLTTRON

VOLTTRON [65,67] is a framework specifically designed for use in electrical power systems. It was developed by the Pacific Northwest National Laboratory (PNNL), and it is available in Python. It is a modular, open source platform that is intended to support transactions between networked elements over the grid.

7.1.4. Aglets

Initially developed at the IBM Tokyo Research Laboratory, Aglets is a mobile agent platform and library that is written in Java [68] that eases the development of agent-based applications. Aglets includes a stand-alone server called Tahiti and a library that enables the developer to build mobile agents, as well as include the Aglets technology within their applications.

7.1.5. JACK

JACK [69,70] is a commercially licensed agent-oriented development environment. It was developed in Java and acts as a Java extension that provides classes for implementing agent behaviour. It provides a graphical user interface for creating agents within projects. It is highly portable and platform independent.

8. MASs with Trust in the Smart Grid

The application of trust within MASs will have a positively impactful role on security within the Smart Grid. However, there has been extremely limited research in this area. The few studies which were identified are mentioned in this section.

8.1. Research Areas

Zhao et al. [71] implemented both direct-based and reputation-based trust mechanisms to create a modified version of the contract net protocol (CNP) [72]. The new trust-based CNP model, which was implemented in distributed MAS architecture, was used in Smart Grid scheduling to ensure improved decision quality which led to improved energy efficiency. With the direct trust mechanism, the time and rating value of the trustee were used to calculate the direct trust. These values are stored individually by each agent. The recommendation trust requires the trust rating of the trustee from all other agents in the MAS. The values generated by the trust mechanisms are fed into the CNP model, which is used to calculate which agent a task is delegated to. The model was tested via simulation using

JADE; therefore, a real-world test was not made. This model has not been tested against trust model-related attacks.

In another paper, an MAS-based negotiation mechanism was implemented to combat jamming attacks in the Smart Grid power market [73]. Their work involved using the trust-based CNP [71] during local marginal price (LMP) [74] negotiations. Their work was simulated on a PJM 5-bus system [75], and it was not tested against any trust-related attacks.

Pereira et al. implemented a trust model in testing the resilience of control systems in power purchasing in cyber-physical systems [76]. The trust is used to calculate the cost of power to be sold by a producer agent to a consumer agent based on the trust level of the consumer. The model was tested using the JADE and GridLab-D power distribution and analysis tool [77].

In another study, trust was used in the secure operation of state estimation algorithms in networked microgrids [78]. Each microgrid within the network was modelled as an agent. Each agent implements direct trust when an agent provides state estimation values that are below a certain threshold. A malicious node is then isolated by the peer agents from the network. The historical data based on which the behaviour of a node was based are not specified, and the tool used for simulation was also not specified. Their work was not tested against trust-related attacks to test its resilience.

Matei et al. [79] proposed a trust-based security mechanism for protecting the state estimation process against false data injection attacks by using a multi-agent filtering scheme. The agents assign a trust metric that is used to disregard messages from low-trusted agents. The mechanism involved a mathematical simulation and was not tested against trust-related attacks. Cunningham et al. [80] wanted to see the impact of trust in a hierarchical agent-based socio-technical system. They ran a scenario replicating the 2003 Northeast Blackout which, in the history of North America, was the largest blackout [81]. The system is comprised of the elements responsible for the handling of the blackout. Each element was identified as an agent. The trust value was a score based on how an agent successfully or unsuccessfully handled a task. Their work was simulated using JADE, and it was not tested against any trust-related attacks. Hussain et al. [82] implemented trust in the inclusion of DERs in Smart Grid. The update of the trust score of an agent was dependent on the adherence to the Service License Agreement between it and other agents. Their work was simulated using the JACK-AOS [83] multi-agent platform and was not tested against trust-related attacks.

Borowski et al. [84] implemented reputation-based trust in an agent-based backup protection scheme that aims to mitigate the effects of faults and faulty agents in substations. Their work was simulated using NS-2 [85], EPOCHS [86] and PSCAD/EMTDC [87] but was not tested against trust-related attacks.

8.2. Discussion

In stark contrast to Section 5, MAS-based trust within the AMI and network communication priority areas do not exist as shown in Table 5. Furthermore, energy storage, electric transportation, and WASA priority areas are still uncharted territories when it comes to trust. There are only three papers each for DER and distribution grid and only two for demand response and consumer energy efficiency areas. Clearly, this shows that a lot of work is required on trust in MAS-based environments in the Smart Grid.

Table 6 shows that the generation, customer and service provider domains have yet to be explored while the markets domain only has two papers. Three papers were focused on the transmission, distribution, and operation domains, while only one was focused on the operations domain and only two focused on only the distribution domain.

Only one paper includes the risk component in its trust model, as shown in Table 7. The knowledge component of most of papers did not include previous transactions or states; therefore, trust was computed based on the values of parameters that were provided for computation. There were three papers that exclusively focused on direct trust, and one paper focused on indirect trust. Five papers focused on both types of trust.

All their works were simulated, and JADE was the most used framework among the tools, as shown in Table 8. Other types of frameworks or applications were used, but they were not discussed because they were not specifically designed for MAS. Six of the papers implemented a decentralized MAS architecture, while three of them implemented a centralized architecture.

Table 5. Research on MAS-based trust in Smart Grid categorized based on NIST priority areas.

NIST Priority Areas							
Distribution Grid Managemen	**Energy Storage**	**AMI**	**Electric Transportation**	**Network Communication**	**Demand Response and Consumer Energy Efficiency**	**WASA**	**DER**
[79,80,84]	-	-	-	-	[71,76]	-	[73,78,82]

Table 6. Research on MAS-based trust in Smart Grid categorized by NIST conceptual domains.

NIST Conceptual Domains						
Transmission	**Generation**	**Distribution**	**Markets**	**Customer**	**Service Provider**	**Operations**
[79,80,84]	-	[78–80,82,84]	[73,76]	-	-	[71,79,80,84]

Table 7. Research on MAS-based trust in Smart Grid categorized by trust components.

Trust Components				
Direct Trust	**Indirect Trust**	**Tested Against Trust Attacks**	**Risk Component**	**Knowledge Component**
[71,73,76,78] [79,80,82,84]	[71,73,76,79,84,88]	-	[76]	[71,73,76,78] [78–80,82]

Table 8. MAS-based trust in Smart Grid categorized by other parameters.

Paper	MAS Architecture	Type of Testing	Tool Used
Zhao et al. [71]	Decentralized	Simulation	JADE
Cintuglu et al. [78]	Decentralized	Simulation	-
Cunningham et al. [80]	Centralized	Simulation	JADE
Alavikia et al. [73]	Decentralized	Simulation	PJM 5-bus system
Matei et al. [79]	Decentralized	Simulation	-
Guemkam et al. [89]	Centralized	Simulation	Utopia, MOISE
Hussain et al. [82]	Centralized	Simulation	Jack-AOS
Borowski et al. [84]	Decentralized	Simulation	JADE, EPOCHS, PPSCAD/EMTDC
Pereira et al. [76]	Decentralized	Simulation	JADE, GridLab-D

9. Motivation

Sections 5 and 6 demonstrate the scarcity of trust-related research within the Smart Grid. Even more so, Section 8 shows the scarcity of trust-related MAS research in the Smart Grid. Trust is essential, especially with respect to communication among IEDs and SCADA. As future work, it would be important for vendors to make IEDs secure-centrically autonomous by encompassing trust to have a security-related impactful role within substations. In the situation of existing IEDs that are resource-constrained, the integration of intelligent agents with IEDs could make this possible.

Trust among devices within the substation must be defined differently. The key parameters required to compute trust within devices are reliant on the communication among devices and SCADA. The type of communication can be a request, command or a response from a device or SCADA. As such, the risks involved in the acceptance of each

communication that is received has to be computed to calculate trust. Furthermore, a history of communications is required to be stored to be used as a reference to compute trust. Concerning the formulation of trust in Section 4.1, trust among IEDs (and also SCADA) can be seen as a tuple with some modifications, as shown in (8).

$$T_{ij} = f(m_{ij}, d_i, d_j, r_{ij}, h_{ij}) \tag{8}$$

m_{ij} is the message being analyzed before it can be trusted and accepted, d_i is the agent device, d_j is the subject device, r_{ij} is the risk involved should the message be accepted or trusted, and h_{ij} is the history of communication between d_i and d_j.

A simple conceptual algorithm is presented in Algorithm 1 where d_i receives m_{ij} from d_j and computes T_{ij} based on m_{ij}. If T_{ij} equals or exceeds the threshold value, m_{ij} is received and acted upon, otherwise it is dropped and an alert is raised. It must be noted that trust can be scaled on a continuum such that certain actions are taken when certain thresholds on that scale are exceeded [90]. Actions can range by sending warnings, raising alarms or in the worst case scenario, refuse to communicate with a non-trusted device.

Algorithm 1 Pseudo-algorithm for trust computation for agent device

```
Receive m_ij
Compute T_ij = f(m_ij, d_i, d_j, r_ij, h_ij)
if T_ij ≥ T_threshold then
    Accept m_ij
else
    Drop m_ij
    Raise alarm
end if
```

In Figure 5, we present a proposed trust model that can be implemented in a substation environment. We define consequence as the measure of damaging impact an action has on a substation. Consequence represents the risk involved when a current action/message is taken within the substation and requires some parameters from familiarity as input. Consequence requires knowing the state of the substation (environment state) and the dependencies (criticality) within the substation to calculate the risk or consequence of the action to be undertaken.

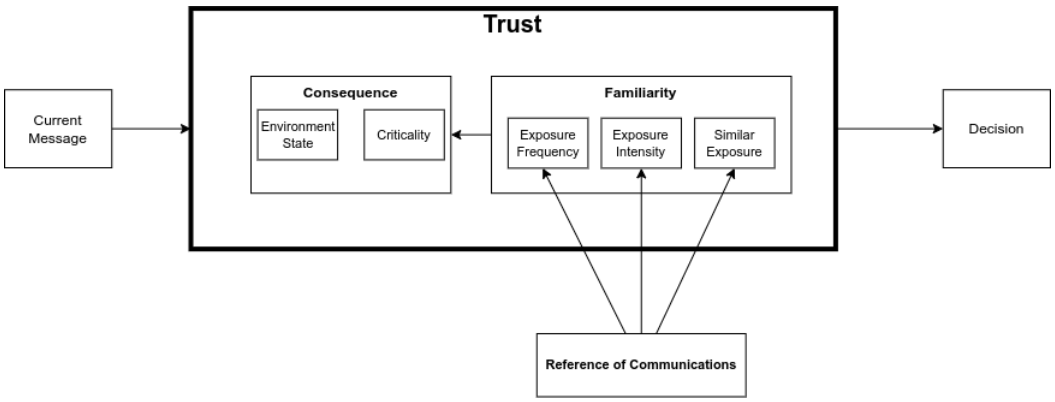

Figure 5. Proposed trust model for substation.

We define familiarity as a measure of the consistency of actions/messages of different types between devices. Familiarity, in this situation, maps to the history of communication or existing knowledge in the trust formalization presented to date. According to Yonelinas [91] and Zhan et al. [92] factors that influence familiarity are exposure intensity, exposure frequency, and similar exposure. Exposure frequency is defined as the frequency with which messages/actions are exposed to the devices.

Exposure intensity is defined as the length of time in which the messages/actions are exposed to the devices. Similar exposure is the measure of the similarity of the messages/actions being exposed to the devices. The mathematical formulation of this model and the results are discussed in the remaining sections this paper.

The environment state is computed using standard computations to ensure fault protection scenarios such as overvoltage, undervoltage, etc. [93]. Computation of the environment is out of the scope of this paper.

10. Criticality

To determine the dependency of devices within the substation, we need to provide a ranking of each device in terms of how critical it is within the substation. The higher the ranking, the higher the cascading effect within the substation. To achieve this, we utilized an artifact from the literature to create the criticality rankings for a substation [94]. According to the paper, for a list of n number of devices, D is defined in Equation (9).

$$D = \{d_1, d_2, \ldots, d_n\};\ 0 \leq i \leq n \tag{9}$$

$$R_{d_i} = \{d_i, d_j, d_{j+1}, \ldots, d_k\};\ \forall\ d_j,\ 0 \leq j, k \leq n \tag{10}$$

$$An_{d_i} = \{d_i, d_j, d_{j+1}, \ldots, d_k\};\ \forall\ d_j,\ 0 \leq i, j, k \leq n \tag{11}$$

$$I_{d_i} = d_{e_i} \cap An_{d_i};\ \forall\ d_i \tag{12}$$

$$l = \{d_i, \ldots, d_k\};\ \forall\ d_i, R_{d_i} = I_{d_i},\ 0 \leq i, k \leq m \tag{13}$$

$$L = \{l_1, l_i, \ldots, l_m\};\ 0 \leq i \leq m \tag{14}$$

For each d_i, a list of devices (including d_i) that are functionally dependent on d_i are generated as shown in Equation (10). The reverse is also performed where the list of devices that functionally influence d_i are also identified as shown in Equation (11). An intersection between R_{d_i} and An_{d_i} is identified using Equation (12).

Within m number of rounds, each d having the same devices in R_{d_i} and I_{d_i} are given similar ranking, l (see Equation (13)). This results in a set of criticality rankings, L as shown in Equation (14). Devices in a single line diagram (Figure 6) were ranked as shown in Table 9 where devices starting with *IED* are the primary focus of this paper and the others can be ignored (details can be found in [94]).

Table 9. Criticality ranks of substation devices.

Level	Devices
Level 9	CB1A, IED1A, CB1B, IED1B
Level 8	IL2, IL1
Level 7	CB1C, DLI1, IED1C, DLI2
Level 6	CB2D, CB2C, IED2D, IED2C, BUS2, BUS1
Level 5	CB3B, CB3A, IED3B, IED3A, DLTB, DLTA
Level 4	CB4B, CB4A, CB4C, CB5B, CB5A, CB6A, CB5C, IED4B, IED4A, IED4C, IED5C, IED5A, IED5B, TXA,TXB
Level 3	CB2B, CB2A, IED2B, IED2A, BUS4, BUS3
Level 2	DL66B, DL66A, OC4C, OC4A, OC4B, OC5B, OC5A, OC5C
Level 1	CT1C, CT1A, CT1B, CT2C, CT2A, CT2D, CT2B, CT3B, CT3A, CT4C, CT4A, CT4B, CT5B, CT5C, CT5A, PT2A, PT2B, PT6A, PT3A, OL2, OL1

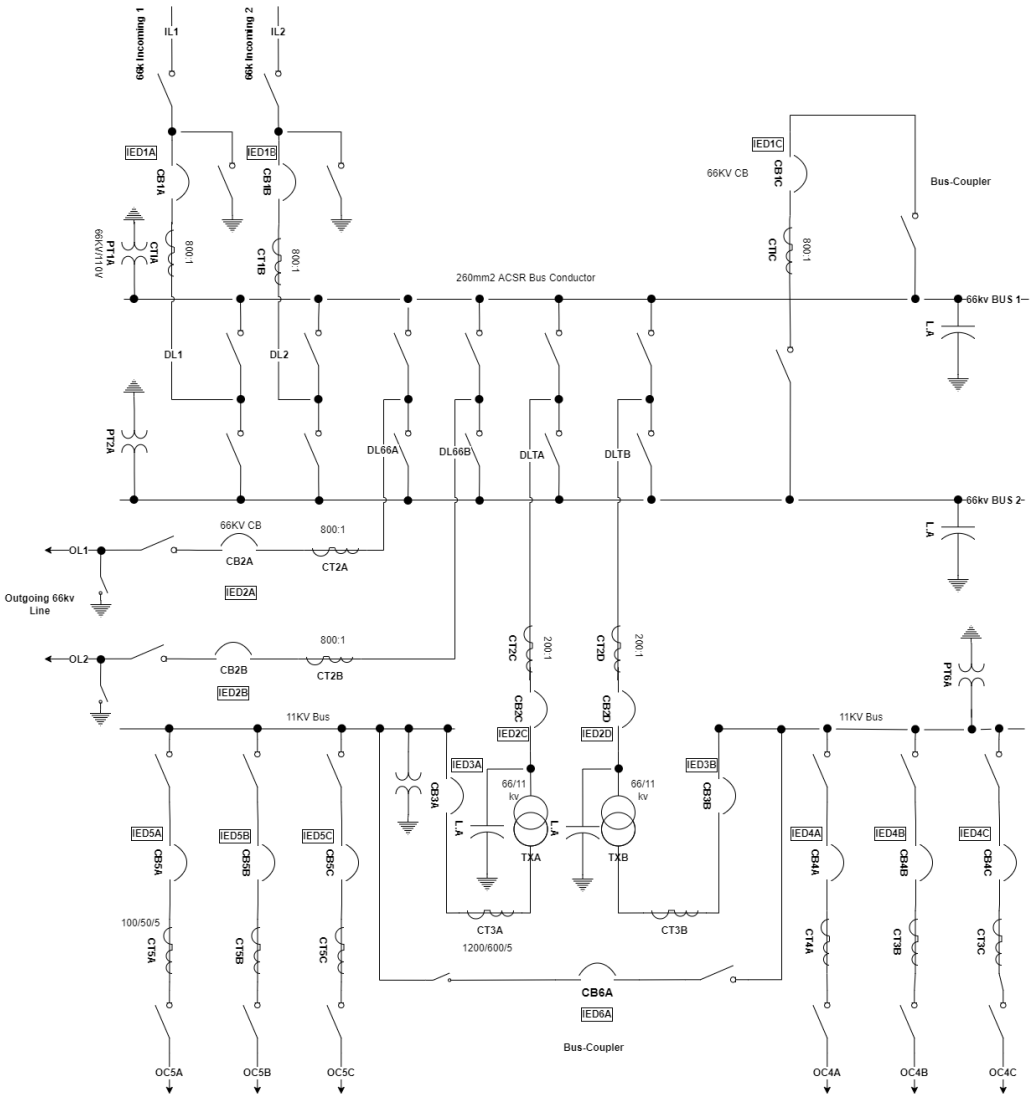

Figure 6. Labelled single-line diagram of 66/11 kV substation (adapted from [95]).

11. Models and Scenario

11.1. Substation Model

We define the substation, Ξ, as a three-tuple entity in Equation (15) where M, S, and N represent sets of clients, servers, and network devices, respectively, (Equations (16)–(18)). N interconnects S and M. There exists a set of queries, Q, and a set of corresponding responses, R, defined in Equations (19) and (20). Periodically, m_i, sends Q to s_i and receives R from s_i. Each m_i–s_i pair may have a unique pair of Q and R. A query and its associated response have either read ($\vartheta = 0$) or write ($\vartheta = 1$) operations. Queries and responses made by the attacker are defined in Equations (21) and (22), respectively.

$$\Xi = (M, N, S) \tag{15}$$

$$M = \{m_0, m_1, \ldots, m_i\} \tag{16}$$

$$S = \{s_0, s_1, \ldots, s_i\} \tag{17}$$

$$N = \{n_0, n_1, \ldots, n_i\} \tag{18}$$

$$Q = \{q_0, q_1, \ldots, q_i\} \tag{19}$$

$$R = \{r_0, r_1, \ldots, r_i\} \tag{20}$$

$$Q' = \{q'_0, q'_1, \ldots, q'_i\} \tag{21}$$

$$R' = \{r'_0, r'_1, \ldots, r'_i\} \tag{22}$$

11.2. Attack Scenarios

With the substation, the ultimate goal of the attacker is gaining control of an element(s) of S to cause an outage within the Smart Grid. In most cases, the IED is that device. We present two scenarios where s_i (or more than one) is compromised.

11.2.1. Compromised Network, A_N

When N is compromised, the attacker, m' or s', sends Q' and/or R' to a device or uses any compromised element in M or S to do so. Unfortunately, there are no publicly recorded incidents of such nature; thus, we use this literature-sourced scenario [96]. In this scenario, A_N, the attacker is oblivious to the substation's architecture and as a result, requires cyber attacks to identify S before transmitting Q' and/or R'. It is assumed that the attacker has already achieved this. Therefore, the possible attacks are identifiable in A_M and below:

- Man-in-the-middle (MitM) attack: m' (or s') impersonates a device to send q' or r';
- Maliciously crafting packets: m' (or s') sends maliciously crafted q' (or r') to drop a payload or trigger a buffer overflow;
- Query flooding: m' (or s') exhausts a device's resources with a bombardment of Q' or R'.

11.2.2. Compromised Client

One notable device in M is SCADA. Publicly available documented attacks of utility companies have identified SCADA as the entry point preceded by successful social engineering attacks. The most notable attacks are Stuxnet, BlackEnergy [97], and Havex [98]. In this scenario, A_M, the attacker controls m_i to become m'_i before transmitting Q'. SCADA's compromise guarantees the attacker an architecture-wide view of the substation. Rarely identified publicly, it is also possible for an attacker to compromise s_i to become s'_i to transmit R'. Thus, the considered attack scenarios are:

- *Reconnaissance*: For $\vartheta = 0$, m'_i transmits q' to s_i to all existing Modbus addresses.
- *Loading Malicious Firmware*: m'_i makes s_i inaccessible by loading a malicious firmware. This can be performed by utilizing a device-specific software within SCADA or embedding malicious bytes in q'. The former option is not within the scope of this paper.
- *Baseline Replay Attack*: m'_i (or s'_i) replays Q or R to a device after profiling the substation to avoid detection.
- *Write attack*: Without reconnaissance and for $\vartheta = 1$, q' is sent to s_i to all existing Modbus addresses. Another scenario requires a completed reconnaissance attack. q', where $\vartheta = 1$, is sent to target an address of a specific s_i. It can be also executed after a baseline replay attack.

11.3. Modbus TCP

Due to its documentation being readily available and it being used by modern and legacy substations (which form a significant percentage of substations worldwide [99]), Modbus TCP [100]—which is the TCP variant of Modbus [101]—is used. Furthermore, reinforcing our selection is the fact that there is current literature that is centred around its security [102], vulnerabilities [103], attack mitigation [104,105], and utilization in testbeds [106,107]. Utilizing TCP port 502, its implementation requires a client–server architecture. Modbus does not support unsolicited responses from servers. The Modbus TCP frame/packet consists of the Modbus Application Header (MBAP) header and the Protocol Data Unit (PDU) with their sizes and those of their components specified in Figure 7.

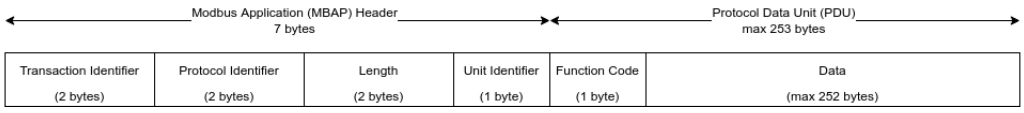

Figure 7. Modbus TCP/IP frame.

The function code determines the request type that is sent to the server and the server responds using the same function code. The address(es) and/or the value written to/read being accessed from the server are specified in the data section of the PDU. The minimum Modbus request size is 12 bytes and that for response is 10 bytes and a maximum of 260 bytes for both. Table 10 shows a selection of the function codes selected for this work based on multiple datasets that were reviewed.

Table 10. Selected modbus function codes.

Address Type	Access Type	Address Size	Function/Query	Function Code (Hex)
Coil	Write/Read	1 bit	Read Coil	0x01
			Write Multiple Coils	0x0F
			Write Single Coil	0x05
Holding Register	Write/Read	2 bytes	Read Holding Register	0x03
			Write Multiple Registers	0x10
			Write Single Register	0x06
Discrete Input	Read	1 bit	Read Discrete Input	0x02
Input Register	Read	2 bytes	Read Input Register	0x04

11.4. Familiarity-Based Definitions

11.4.1. Exposure Intensity

When q_i or r_i is transmitted, a set of features, Z (Equation (23)), is created and used to compute exposure intensity, E_i, as shown in Equation (29), where $E_i \rightarrow [0, 1]$. An alert description, κ_{E_i}, associated with the value of E_i. The description of each feature is available in the table of notations. The sender's current message's arrival time, t_i, the sender's previous message's arrival time, t_{i-1}, the sender's first message's arrival time, t_0, the sender's last message's arrival time, t_n, and the recipient's dispatched message's time, t_{d_i}, are required to define the features in Equations (24)–(28).

$$Z = \{\zeta_{pt}, \zeta_{qq}, \zeta_{qr}, \zeta_{tt}, \zeta_{to}\} \tag{23}$$

$$\zeta_{qq} \;||\; \zeta_{rr} = t_i - t_{i-1} \tag{24}$$

$$\zeta_{qr} = t_i - t_{d_i} \tag{25}$$

$$\zeta_{tt} = \begin{cases} 0, \texttt{ if } i = q0 \\ t_i - t_0, \texttt{ if otherwise} \end{cases} \tag{26}$$

$$\zeta_{to} = \begin{cases} t_n - t_i, \texttt{ if } i = 0 \\ 0, \texttt{ if otherwise} \end{cases} \tag{27}$$

$$\zeta_{pt} = \begin{cases} +1, \texttt{ if } \zeta_{qq} < \zeta_{qq}^T, \\ +1, \texttt{ if } \zeta_{rr} < \zeta_{rr}^T, \\ +1, \texttt{ if } \zeta_{qr} < \zeta_{qr}^T, \\ +1, \texttt{ if } \zeta_{qq} < \zeta_{qr}, \\ +1, \texttt{ if } \zeta_{rr} < \zeta_{qr}, \\ +1, \texttt{ if } \zeta_{to} < \zeta_{to}^T, \\ 0, \texttt{ if otherwise} \end{cases} \tag{28}$$

$$E_i = \begin{cases} 1, \texttt{ if } \vartheta = 1 \\ 0, \texttt{ if } \zeta_{pt} > \zeta_{pt}^T, \kappa_{E_i} = 1 \\ \frac{Z_R \cdot Z_i}{\|Z_R\| \|Z_i\|}, \kappa_{E_i} = 0 \texttt{ if } E_i > E_i^T \\ \frac{Z_R \cdot Z_i}{\|Z_R\| \|Z_i\|}, \kappa_{E_i} = 3 \texttt{ if } E_i < E_i^T \end{cases} \tag{29}$$

11.4.2. Similar Exposure

When q_i or r_i is transmitted, a Moore machine is defined in Equation (30).

$$\Upsilon = \{\rho, \sigma, \delta, \rho_0, \Psi, \lambda\} \tag{30}$$

When m_i transmits Q to s_i, a Moore machine, $\Upsilon = \{\rho, \sigma, \delta, \rho_0, \Psi, \lambda\}$, is defined to parse through q_i as follows (the definition of each symbol can be found in the table of notations):

- ρ defined in Equation (31) represents a set of states where each state represents q_i or r_i where ρ_0 is the initial state. Accept states are not required due endless transmissions of q_i or r_i.
- σ, defined in Equation (32), is a set of input alphabets extracted from q_i or r_i.
- δ is the transition function defined in Equation (33).
- A set of features, Ψ, is an output of λ (Equation (36)).
- The output function, λ, is defined in Equation (34) which is the output function that maps ρ to Ψ. Equations (35)–(39) define the mappings.

$$\rho = \{\rho_{rdi}, \rho_{wsc}, \rho_{rc}, \rho_{wmc}, \rho_{wsr}, \rho_{rhr}, \rho_{wmr}, \rho_u, \rho_{rir}\} \tag{31}$$

$$\sigma = \begin{cases} fc_{q_i} | a_{q_i} | \iota_{q_i}, \texttt{ if } \vartheta = 0 \\ fc_{q_i} | a_{q_i}, \texttt{if } \vartheta = 1 \\ fc_{r_i} | b_{r_i} | \iota_{r_i} \,||\, fc_{r_i} | a_{r_i}, \texttt{ if } \vartheta = 1 \\ fc_{r_i} | b_{r_i}, \texttt{ if } \vartheta = 0 \end{cases} \tag{32}$$

$$\delta : \rho \times \sigma \rightarrow \rho \tag{33}$$

$$\lambda : \rho \rightarrow \Psi \tag{34}$$

$$\rho_{rdi} : \psi_s = 1, \psi_{ma} = 1, \psi_{fc} = 1, \psi_{mas} = 1, \psi_{mdiq} = 1, \psi_{mdir} = 1 \tag{35}$$

$$\Psi = \{\psi_s, \psi_p, \psi_\eta, \psi_{us}, \psi_{mas}, \psi_{ma}, \psi_{fc}, \psi_{mdiq}, \psi_{mdir}, \psi_{mcr}, \psi_{mhrr}, \psi_{mcq}, \psi_{mhrq}, \psi_{mirq}, \psi_{mirr}\} \tag{36}$$

$$\rho_{rc}, \rho_{wmc}, \rho_{wsc} : \psi_s = 1, \psi_{ma} = 1, \psi_{fc} = 1, \psi_{mas} = 1, \psi_{mcq} = 1, \psi_{mcr} = 1 \tag{37}$$

$$\rho_{rir} : \psi_s = 1, \psi_{ma} = 1, \psi_{fc} = 1, \psi_{mas} = 1, \psi_{mirq} = 1, \psi_{mirr} = 1 \tag{38}$$

$$\rho_u : \psi_{us} = 1 \tag{39}$$

$$\rho_{wmr}, \rho_{wsr}, \rho_{rhr} : \psi_{mas} = 1, \psi_s = 1, \psi_{ma} = 1, \psi_{fc} = 1, \psi_{mirr} = 1, \psi_{mhrr} = 1 \tag{40}$$

A set of IP-MAC pairs, H (Equation 41), is required for the definition of ψ_η in Equation (42).

$$H = \{\eta_0, \eta_1, \ldots, \eta_i\} \tag{41}$$

$$\psi_\eta = 1, \texttt{ if } \eta \notin H \tag{42}$$

Finally, E_s is defined in Equation (43) where $E_i \rightarrow [0, 1]$ and based on the generated value, the associated κ_{E_s} is generated.

$$E_s = \begin{cases} 0, \texttt{ if } \psi_{\eta_i}, \kappa_{E_s} = 1 \\ 0, \texttt{ if } \psi_{p_i} \neq 502, \kappa_{E_s} = 2 \\ \frac{\Psi_R \cdot \Psi_i}{\|\Psi_R\| \|\Psi_i\|}, \kappa_{E_s} = 0, \texttt{ if } E_s > E_s^T \\ \frac{\Psi_R \cdot \Psi_i}{\|\Psi_R\| \|\Psi_i\|}, \kappa_{E_s} = 3, \texttt{ if } E_s < E_s^T \end{cases} \tag{43}$$

11.4.3. Exposure Frequency

For each q_i or r_i that is received, Γ is defined in Equation (44).

$$\Gamma = \{\gamma_{fs}, \gamma_{frc}, \gamma_{cq}, \gamma_{frdi}, \gamma_{diq}, \gamma_{frhr}, \gamma_{hrq}, \gamma_{frir}, \gamma_{irq}, \gamma_{fwsc}, \gamma_{cv}, \gamma_{fwsr}, \gamma_{hrv}, \gamma_{fwmc}, \gamma_{cvs}, \gamma_{fwmr}, \gamma_{hrvs}, \gamma_{cdc}, \gamma_{didc}, \gamma_{irdc}, \gamma_{hrdc}, \gamma_{divs}, \gamma_{irvs}\} \quad (44)$$

For each q_i or r_i, each feature is defined as follows:

- For $fc_{q_i} = 1$, $\gamma_{frc} = 1$, $\gamma_{cq} = \iota_{q_i}$.
- For $fc_{q_i} = 2$, $\gamma_{frdi} = 1$, $\gamma_{diq} = \iota_{q_i}$.
- For $fc_{q_i} = 3$, $\gamma_{frhr} = 1$, $\gamma_{hrq} = \iota_{q_i}$.
- For $fc_{q_i} = 4$, $\gamma_{frir} = 1$, $\gamma_{irq} = \iota_{q_i}$.
- For $fc_{q_i} = 5$, $\gamma_{fwsc} = 1$, $\gamma_{cv} = d_{q_i}$, $\gamma_{cdc} = 2$, $\gamma_{cq} = 1$.
- For $fc_{q_i} = 6$, $\gamma_{fwsr} = 1$, $\gamma_{hrv} = d_{q_i}$, $\gamma_{hrdc} = 2$, $\gamma_{hrq} = 1$.
- For $fc_{q_i} = 15$, $\gamma_{fwmc} = 1$, $\gamma_{cvs} = d_{q_i}$, $\gamma_{cdc} = b_{q_i}$, $\gamma_{cq} = \iota_{q_i}$.
- For $fc_{q_i} = 16$, $\gamma_{fwmr} = 1$, $\gamma_{hrvs} = d_{q_i}$, $\gamma_{hrdc} = b_{q_i}$, $\gamma_{hrq} = \iota_{q_i}$.
- For $fc_{r_i} = 1$, $\gamma_{frc} = 1$, $\gamma_{cdc} = b_{r_i}$, $\gamma_{cvs} = d_{r_i}$
- For $fc_{r_i} = 2$, $\gamma_{frdi} = 1$, $\gamma_{didc} = b_{r_i}$, $\gamma_{divs} = d_{r_i}$
- For $fc_{r_i} = 3$, $\gamma_{frhr} = 1$, $\gamma_{hrdc} = b_{r_i}$, $\gamma_{hrvs} = d_{r_i}$
- For $fc_{r_i} = 4$, $\gamma_{frir} = 1$, $\gamma_{irdc} = b_{r_i}$, $\gamma_{irvs} = d_{r_i}$
- For $fc_{r_i} = 5$, $\gamma_{fwsc} = 1$, $\gamma_{cv} = d_{r_i}$, $\gamma_{cdc} = 2$, $\gamma_{cq} = 1$.
- For $fc_{r_i} = 6$, $\gamma_{fwsr} = 1$, $\gamma_{hrv} = d_{r_i}$, $\gamma_{hrdc} = 2$, $\gamma_{hrq} = 1$.
- For $fc_{r_i} = 15$, $\gamma_{fwmc} = 1$, $\gamma_{cq} = \iota_{r_i}$.
- For $fc_{r_i} = 16$, $\gamma_{fwsr} = 1$, $\gamma_{hrq} = \iota_{r_i}$.
- $\gamma_{fs} = l_d$

Exposure frequency, E_f, is finally defined in Equation(45)—where $E_f \rightarrow [0, 1]$ and the corresponding κ_{E_f} is generated.

$$E_f = \begin{cases} 0, \texttt{ if } l_{h_i} < 7, \kappa_{E_f} = 1 \\ 0, \texttt{ if } \gamma_{fs_R} \neq \gamma_{fs_i}, \kappa_{E_f} = 2 \\ 0, \texttt{ if } \gamma_{fc_i} = 0, \kappa_{E_f} = 3 \\ \frac{\Gamma_R \cdot \Gamma_i}{\|\Gamma_R\| \|\Gamma_i\|}, \kappa_{E_f} = 0 \texttt{ if } E_f > E_f^T \\ \frac{\Gamma_R \cdot \Gamma_i}{\|\Gamma_R\| \|\Gamma_i\|}, \kappa_{E_f} = 4 \texttt{ if } E_f < E_f^T \end{cases} \quad (45)$$

11.4.4. Familiarity

Using all the exposures, we define familiarity, F_i, Equation (46) where $F_i \not> min\{E_f, E_s, E_i\}$ and $F_i \rightarrow [0, 1]$.

$$F_i = \frac{2}{\sqrt{2}} \begin{vmatrix} \sqrt{\frac{1}{2}}E_f & \sqrt{\frac{1}{2}}E_f & 0 & 1 \\ 0 & \sqrt{\frac{1}{2}}E_s & \sqrt{\frac{1}{2}}E_s & 1 \\ \sqrt{\frac{1}{2}}E_i & 0 & \sqrt{\frac{1}{2}}E_i & 1 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (46)$$

11.5. Consequence-Based Definitions

In determining consequence-related values, any q_i or r_i, where $\vartheta = 1$ is transmitted within a non-permitted time or scenario in a value of 1. For scenarios or periods where $\vartheta = 0$, the ratio of the criticality of the device (see Equation (14)) to the highest criticality ranking, ϱ, is used unless in exceptional cases.

$$\varrho_i = \frac{l_i}{l_m} \quad (47)$$

φ'_{ω}, φ_{ω}, φ'_{χ}, φ_{χ}, and φ_{ξ} are sensitivity weights for adjusting ϱ such that $\varphi \rightarrow [0,1]$ and $\varrho \rightarrow [0,1]$.

11.5.1. Environment Status Attack Value

The function $E(p) \rightarrow \{0,1\}$ determines the p's state—and is a substation property. The environment flag, τ, is evaluated as shown in Equation (48)

$$\tau = \begin{cases} 0, \texttt{ if } \vartheta | \vartheta = 1, E(p) = 1, \\ 1, \texttt{ if } \vartheta | \vartheta = 1, E(p) = 0, \kappa_{C_\tau} = 1 \end{cases} \tag{48}$$

11.5.2. Replay Attack Value

Here, the count of replay, y, increases by 1 if $q_i = q_{i-1}$ or $r_i = r_{i-1}$. Therefore, using y, with y^T being its threshold, the replay attack value, ω, is calculated in Equation (49).

$$\omega = \begin{cases} \varrho_i * \varphi'_{\omega}, \texttt{ if } y >= 1, \psi_{us} >= 1, \kappa_{C_\omega} = 5, \vartheta = 0 \\ \varrho_i * \varphi_{\omega}, \texttt{ if } y >= 1, \kappa_{C_\omega} = 1, \vartheta = 0 \\ 1, \texttt{ if } y > \vartheta, \psi_{us} >= 1, \kappa_{C_\omega} = 6 \\ 1, \texttt{ if } y >= 1, \kappa_{C_\omega} = 2, \vartheta = 1 \\ 1, \texttt{ if } y >= 1, \psi_{us} >= 1, \kappa_{C_\omega} = 4, \vartheta = 1 \\ 1, \texttt{ if } y > y^T, \kappa_{C_\omega} = 3 \\ 0, \texttt{ if } otherwise \end{cases} \tag{49}$$

11.5.3. Reconnaissance Attack Value

Using ι, ι^{max} and ι^T, for any q_i or r_i, the reconnaissance ranking, ξ, is described in Equation (50).

$$\xi = \begin{cases} 1, \texttt{ if } \iota = \iota^{max}, \kappa_{C_\xi} = 6 \\ \varrho_i * \varphi_{\xi}, \texttt{ if } \iota^T < \iota < \iota^{max}, \vartheta = 0, \kappa_{C_\xi} = 1 \\ 1, \texttt{ if } \psi_{us} > \psi_{us}^T, \kappa_{C_\xi} = 4 \\ 1, \texttt{ if } \iota^T < \iota, \vartheta = 1, \kappa_{C_\xi} = 5 \\ 1, \texttt{ if } 0 < \psi_{us} < \psi_{us}^T, \vartheta = 1, \kappa_{C_\xi} = 3 \\ \varphi_{\xi}, \texttt{ if } 0 < \psi_{us} < \psi_{us}^T, \vartheta = 0, \kappa_{C_\xi} = 2 \\ 0, \texttt{ if } otherwise \end{cases} \tag{50}$$

11.5.4. Query Flooding Attack Value

Utilizing ψ_{us} and ζ_{pt}, the query flooding rating, χ, is calculated in Equation (51)

$$\chi = \begin{cases} 1, \texttt{ if } \zeta_{pt} > \zeta_{pt}^T, \psi_{us} > \psi_{us}^T, \kappa_\xi = 1 \\ \varrho_i * \varphi'_{\chi}, \texttt{ if } \zeta_{pt} > 1, \psi_{us} > 1, \vartheta = 0, \kappa_\xi = 2 \\ \varrho_i * \varphi_{\chi}, \texttt{ if } \zeta_{pt} > 1, \vartheta = 0, \kappa_\xi = 3 \\ 1, \texttt{ if } \zeta_{pt} > 1, \psi_{us} > 1, \vartheta = 1, \kappa_\xi = 4 \\ 0, \texttt{ if } otherwise \end{cases} \tag{51}$$

11.5.5. Packet Manipulation Attack Value

Using l_f and id, for any q_i or r_i, the score of the datagram manipulation, ϕ, is estimated in Equation (52)

$$\phi = \begin{cases} 1, \texttt{ if } id_{r_i} \neq id_{q_i} \\ 1, \texttt{ if } l_{f_{q_i}} \leq 12 \\ 1, \texttt{ if } l_{f_{r_i}} \leq 10 \\ 1, \texttt{ if } 260 \leq l_f \\ 1, \texttt{ if } \gamma_{cv} \neq 0 \,||\, \gamma_{cv} \neq \texttt{0xFF00 for } fc_{q_i} = 5 \\ 1, \texttt{ if } 2\gamma_{cq} \neq \gamma_{cdc} \texttt{ for } fc_{q_i} = 15 \\ 1, \texttt{ if } \gamma_{cdc} > \gamma_{cdc}^{T} \texttt{ for } fc_{q_i} = 15 \\ 1, \texttt{ if } \gamma_{cdc} \neq |\gamma_{cvs}| \texttt{ for } fc_{q_i} = 15 \,||\, fc_{r_i} = 1||2 \\ 1, \texttt{ if } \frac{1}{8}\gamma_{hrq} \neq \gamma_{rhdc} \texttt{ for } fc_{q_i} = 16 \\ 1, \texttt{ if } \frac{1}{8}\gamma_{hrq} + 1 \neq \gamma_{rhdc} \texttt{ for } fc_{q_i} = 16 \\ 1, \texttt{ if } \gamma_{rhdc} > \gamma_{rhdc}^{T} \texttt{ for } fc_{q_i} = 16 \\ 1, \texttt{ if } \gamma_{rhdc} \neq |\gamma_{hrvs}| \texttt{ for } fc_{q_i} = 16 \\ 1, \texttt{ if } \gamma_{rhdc} \neq |\gamma_{hrvs}| \texttt{ for } fc_{r_i} = 4 \,||\, 3 \\ 0, \texttt{ if } otherwise \end{cases} \tag{52}$$

11.5.6. Consequence

Applying the use of τ, ω, xi, χ, and phi, the consequence, C_i, is calculated in Equation (53)

$$C_i = \begin{cases} 0, \texttt{ if } \tau|\omega|xi|\chi|phi = 0, \kappa_C = 0 \\ \xi, \texttt{ if } \xi \neq 0, \kappa_C = \kappa_\xi \\ \tau, \texttt{ if } \tau \neq 0, \kappa_C = \kappa_\tau \\ \omega, \texttt{ if } \omega \neq 0, \kappa_C = \kappa_\omega \\ \chi, \texttt{ if } \chi \neq 0, \kappa_C = \kappa_\chi \\ \phi, \texttt{ if } \phi \neq 0, \kappa_C = \kappa_\phi \end{cases} \tag{53}$$

11.6. Trust

Trust, T_i in Equation (54), is prescribed as an ordered set of values (tuple) with β_i as the score of the trust. The values of κ describe what negatively altered trust. β_i is interpreted in Equation (55), where $\beta_i \rightarrow [-1, 1]$, θ_I is the original state prior to the calculation of trust, β_i^o is the score of the previous trust, β_i^T is the threshold of the trust score, μ represents the weight of forgiveness where $\mu \rightarrow [0, 1]$, and θ_μ represents the condition/state of the forgiveness. The attributes of forgiveness are deferred for later works. r_{ij} in Equation (4) maps to C_i, T'_{ij} maps to β_i^o, and the additional parameters linked to the three exposures. This is primarily due to the inherent information these exposures contain about those parameters.

$$T_i = \{\beta_i, \kappa_{E_f}, \kappa_{E_s}, \kappa_{E_i}, \kappa_C\} \tag{54}$$

$$\beta_i = \begin{cases} \beta^T, \texttt{ if } \theta_I = 1 \\ F_i - C_i, \texttt{ if } \theta_I = 0 \\ F_i - C_i + \mu, \texttt{ if } \theta_I = 0, \theta_\mu = 1, \beta_i^o < \beta^T \end{cases} \tag{55}$$

12. Implementation

Prior to our model assessment, assumptions made were as follows:

- The network communication of this substation is predictable because Q is pre-set by engineers.
- The pristineness of this substation; therefore, $\vartheta = 1$ queries will be considered as malicious.

- The existence of a determinate number of devices inside the network of the substation for the Modbus communication; hence, H, is additionally bounded. These pairs can be categorised into two: the client group, H_m, and the server group, H_s. Additionally, H_s is restricted from sending arbitrary responses. IP–MAC pairs outside this group are considered malicious and grouped as H_a.
- Attacks that are neither Modbus nor IT-related are publicly disclosed by numerous CVE and CWE mitigation techniques; accordingly, they are considered outside of the sphere of the undertaking in this paper.
- The networking port utilized for Modbus communication by a device is restricted to the port number stated in the Modbus specification document.
- The attacker has penetrated the substation, achieved persistence, and has successfully evaded detection.

Datasets with both malicious and normal traffic were critical in our ability to effectively test our suggested model. The EPM dataset was one of two datasets that met our requirements [108] and the other being the ATENA H2020 dataset [109]. We took the following steps:

- The reference features (Equations (23), (36) and (44)) for the exposures in Section 11.4 were generated using the benign traffic captures of the two datasets.
- Based on established documentation of the datasets and careful analysis of every network capture file (pcap file) using Wireshark, H_m, H_s, and H_a could be identified.
- From H_m and H_s, members that were compromised were grouped as H'. The rest of the members were the target devices, H_t.
- Per each dataset, we concentrated on communications that were concerned with H_t and generated sub-capture files containing their communication with the other groups.

Because H' was limited in the datasets, we relied on three types of tests to cover the attack scenarios (see Section 11.2) mentioned in this paper:

- External Attack Test: Here, the existing condition is maintained as H_m, H_s, and H_a; hence, H_a complies with the attack scenario A_N mentioned in Section 11.2.1. Evidently, the outcome is that Q' or R' sent from H_a will be flagged as expected without probing into the Modbus frame (see the first definition of Equation (43)).
- Internal Attack Test: For this test, we have H'_m (Equation (56)) and H'_s (Equation (57)) to depict A_M as described in Section 11.2.2. Any r'_i or q'_i sent from these groups be flagged accordingly.

$$H'_m : H_m \cup H_a \cup H' \tag{56}$$

$$H'_s : H_s \cup H_a \cup H' \tag{57}$$

- Internal Attack Test with IP-MAC Blacklisting: The test and groups are the same as the internal test with the exception that any device that has $\beta_i < \beta^T$ is added to a group of blacklisted MAC-IP pairs, H_b; and is closed from further communication.

A Java application of the trust model was built to test the generated sub-pcap file. We used pcap4j library [110] to parse the Modbus packets. We then mapped a trust scale from the literature [90] to the Multi-State Information Sharing and Analysis Center (MS-ISAC) Alert Information [111] (Figure 8). The threshold flags for the exposures were set to 0.6 and $\beta^T = 0$. All three tests were implemented for trust computation on the server side because there were external devices and internal devices that were used as attackers. Only the internal test was implemented for trust computation on the server side because there were only internal devices that were used for attack. Furthermore, the IP-MAC blacklist was well demonstrated on the server-side test so presenting it in this paper was deemed redundant.

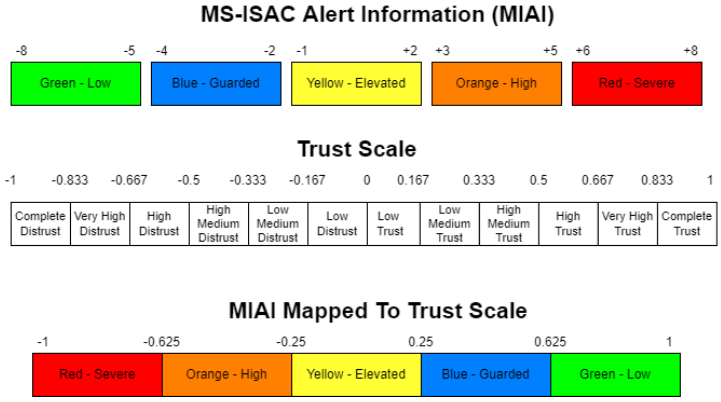

Figure 8. MS-ISAC alert mapped to trust scale [90,111].

13. Evaluation

This section discusses the implementation results in Section 12. An abridged description of each dataset is given before the discussion of the results. A summary of the trust score and alert descriptions/flags for sub-captures with unique characteristics are provided due to page limitations.

13.1. EPM Dataset

The Modbus dataset was used to test our work and the convert channel dataset of the EPM dataset was ignored [108]. This dataset has six benign capture files and five capture files that contain both benign and malicious traffic. The following attacks were implemented in the dataset. These were reconnaissance (characterization), command-and-control, moving malicious files, sending fake commands, and exploits. With the exception of reconnaissance, all other attacks were labelled; thus, we were able to provide the percentage of attacks detected by our model for those attacks.

We were able to do all three kinds of test from the server-side point of view. However, for the client-side point of view, only the internal tests were done because by default, clients do not send queries to external entities. Furthermore, the tests were only done on the command-and-control and moving malicious files attacks because those which involved a client device but the other attacks did not. On the server-side, we identified all malicious client-side messages and we did the reverse for the client-side. Tables 11 and 12 showed that our model detected all the labelled attacks. We will explain our observations and delve deeper into the following sections.

Table 11. Detected attacks towards the server.

	Server				
Attack	**Labelled Packets**	**External**	**Percentage (External)**	**Internal**	**Percentage (Internal)**
CNC	76	76	100%	76	100%
Exploit	780	780	100%	780	100%
Moving Files	39	39	100%	39	100%
Send Fake Command	6	6	100%	6	100%

Table 12. Detected attacks towards the client.

	Client				
Attack	**Labelled Packets**	**External**	**Percentage (External)**	**Internal**	**Percentage (Internal)**
CNC	11	11	100%	11	100%
Moving Files	17	17	100%	17	100%

13.1.1. External Attack Test towards Server

All packets sent from a member of H' were flagged as an *IP-MAC Mismatch*; thus, they were also assigned a *Complete Distrust* score and a *Red—Severe* alert level (Figure 9a). Figure 9b shows whether the packet is Modbus-related (Q or Q') or not. These attacks affected E_s.

In scenarios where attacks were from a member of H_m, the model flagged them and gave the appropriate scores. In the reconnaissance attack, a member of H_m sent packets using a non-Modbus port that were flagged accordingly and assigned *Complete Distrust* scores (Figure 10. This attack also affected E_s. Assuming that the substation environment was in a normal state, any q where $\vartheta = 1$ was flagged as an APT threat, as displayed in Figure 11—it also affected E_f .

trust_score	risk_level	trust_level	count
-1.000000	Red - Severe	Complete Distrust	65

(**a**) Alert/TRUST Score Summary.

Alert	Count
SE: IP-MAC Mismatch-Modbus | EF: Not Computed | EI: Not Computed | CONS: Not Computed	56
SE: IP-MAC Mismatch | EF: Not Computed | EI: Not Computed | CONS: Not Computed	9

(**b**) Alert Details Summary.

Figure 9. Server: EPM dataset external test-H' in CNC capture.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	21
0.99	Green - Low	Complete Trust	77
0.91	Green - Low	Complete Trust	1
0.98	Green - Low	Complete Trust	2
0.96	Green - Low	Complete Trust	1
-1.00	Red - Severe	Complete Distrust	10

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	102
SE: Port Mismatch | EF: Not Computed | EI: Not Computed | CONS: Not Computed	10

(**b**) Alert Details Summary.

Figure 10. Server: EPM dataset external test-H_m in characterization attack capture.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	45
0.99	Green - Low	Complete Trust	119
0.96	Green - Low	Complete Trust	1
-1.00	Red - Severe	Complete Distrust	3
0.87	Green - Low	Complete Trust	1
0.94	Green - Low	Complete Trust	22
0.81	Green - Low	Very High Trust	14
0.98	Green - Low	Complete Trust	2

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	204
SE: Normal | EF: Normal | EI: Normal | CONS: Environment Attack. APT Threat	3

(**b**) Alert Details Summary.

Figure 11. Server: EPM external test-H_m in send-fake-command attack capture.

13.1.2. Internal Attack Test towards Server

For exploiting moving files and CnC (Figure 12) attacks, all packets from a member of H' were given a *Complete Distrust* score and *Red—Severe* alert level which affected E_s. Characterization, however, showed a member of H' sent and replayed Q' which were given a *Low Medium Distrust* score and *Yellow - Elevated* alert level, as shown in Figure 13. In the send-fake-command scenario, a member of H' sends a write request and is flagged accordingly, as shown in Figure 14.

Trust Score	Alert Level	Trust Level	Packet Count
-1.00	Red - Severe	Complete Distrust	65

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Port Mismatch | EF: Not Computed | EI: Not Computed | CONS: Not Computed	65

(**b**) Alert Details Summary.

Figure 12. Server: EPM dataset internal test-H' in command-and-control attack capture.

Trust Score	Alert Level	Trust Level	Packet Count
-0.20	Yellow - Elevated	Low Medium Distrust	3345

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Below Threshold | EF: Normal | EI: Normal | CONS: Unknown Read Query	3342
SE: Below Threshold | EF: Normal | EI: Normal | CONS: Replay of Unknown Read Query|Unknown Read Query|	3

(**b**) Alert Details Summary.

Figure 13. Server: EPM dataset internal test-H' in characterization attack capture.

Trust Score	Alert Level	Trust Level	Packet Count
-1.00	Red - Severe	Complete Distrust	1

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Below Threshold | EF: Normal | EI: Normal | CONS: Unknown Write Query Attack	1

(**b**) Alert Details Summary.

Figure 14. Server: EPM dataset internal test-H' in send-fake-command attack scenario.

13.1.3. Internal Attack Test with IP-MAC Blacklisting towards Server

For the attacks, when the trust score of q is below the threshold, a member of H' becomes a member of H_b and that is visible for exploit, CnC and moving-file captures when an unknown port was used (Figure 15). This means that all kinds of packets bearing the blacklisted member's IP and MAC addresses were dropped; thus explaining the relatively larger number of flagged packets. The same goes for a send-fake-command scenario as well (Figures 16 and 17).

Trust Score	Alert Level	Trust Level	Packet Count
-1.00	Red - Severe	Complete Distrust	781

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Port Mismatch | EF: Not Computed | EI: Not Computed | CONS: Not Computed	1
SE: Not Computed | EF: Not Computed | EI: Not Computed | CONS: Blacklisted IP-MAC	780

(**b**) Alert Details Summary.

Figure 15. Server: EPM dataset internal test with IP-MAC blacklisting-H' in exploit attack capture.

Trust Score	Alert Level	Trust Level	Packet Count
-1.00	Red - Severe	Complete Distrust	4

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Below Threshold | EF: Below Threshold | EI: Normal | CONS: Coil Value Mismatch|Unknown Write Query Attack|	1
SE: Not Computed | EF: Not Computed | EI: Not Computed | CONS: Blacklisted IP-MAC	3

(**b**) Alert Details Summary.

Figure 16. Server: EPM dataset internal test with IP-MAC blacklisting-H' in send-fake-command attack capture.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	3
0.99	Green - Low	Complete Trust	14
0.96	Green - Low	Complete Trust	1
-1.00	Red - Severe	Complete Distrust	943

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	18
SE: Normal | EF: Below Threshold | EI: Normal | CONS: Coil Value Mismatch|Environment Attack. APT Threat|	1
SE: Not Computed | EF: Not Computed | EI: Not Computed | CONS: Blacklisted IP-MAC	942

(**b**) Alert Details Summary.

Figure 17. Server: EPM dataset internal test with IP-MAC blacklisting-H_m in send-fake-command attack capture.

13.1.4. Internal Attack Test towards Client

Attack scenarios where actual client devices were used are CnC and moving-files attacks. As such, those were the captures that we used to test our model. It can be shown in Figures 18 and 19 that malicious packets used a different port and as such were flagged by the model accordingly.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	23
0.99	Green - Low	Complete Trust	1
-1.00	Red - Severe	Complete Distrust	10

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	24
SE: Port Mismatch | EF: Not Computed | EI: Not Computed | CONS: Not Computed	10

(**b**) Alert Details Summary.

Figure 18. Client: EPM dataset internal test-H_s in command-and-control capture.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	60
-1.00	Red - Severe	Complete Distrust	17

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	60
SE: Port Mismatch | EF: Not Computed | EI: Not Computed | CONS: Not Computed	17

(**b**) Alert Details Summary.

Figure 19. Client: EPM dataset internal Test-H_s in moving files capture.

We modified the reference feature values for one of the requests to test the model's output. The model flagged the 'unknown' requests with *High Medium Trust* score and a *Blue—Guarded* alert level (Figures 20 and 21).

Trust Score	Alert Level	Trust Level	Packet Count
0.91	Green - Low	Complete Trust	8
0.50	Blue - Guarded	High Medium Trust	16
-1.00	Red - Severe	Complete Distrust	10

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	8
SE: Normal | EF: Below Threshold | EI: Normal | CONS: Normal	16
SE: Port Mismatch | EF: Not Computed | EI: Not Computed | CONS: Not Computed	10

(**b**) Alert Details Summary.

Figure 20. Client: EPM dataset internal test-H_s in command-and-control capture with modified reference features.

Trust Score	Alert Level	Trust Level	Packet Count
0.91	Green - Low	Complete Trust	20
0.50	Blue - Guarded	High Medium Trust	38
-1.00	Red - Severe	Complete Distrust	17
0.48	Blue - Guarded	High Medium Trust	1
0.49	Blue - Guarded	High Medium Trust	1

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	20
SE: Normal | EF: Below Threshold | EI: Normal | CONS: Normal	40
SE: Port Mismatch | EF: Not Computed | EI: Not Computed | CONS: Not Computed	17

(**b**) Alert Details Summary.

Figure 21. Client: EPM dataset internal test-H_s in moving files capture with modified reference features.

13.2. ATENA H2020 Dataset

In this dataset, ICMP flooding, TCP SYN flooding, Modbus query flooding, and MitM attacks were implemented. Of the four attacks, Modbus queryflooding and MitM attacks were focused more on Modbus. Regardless, capture files involving all four attacks had some Modbus packets in there so we focused on those. The dataset was grouped into three sets of capture files. The length of each capture file was either 30 min, 1 h, or 6 h. The attack duration was in series of either 1, 5, 15, or 30 min. We observed that only one read-access function code was implemented in this dataset; thus, we deactivated Equation (49) for consequence.

13.2.1. External Attack Test towards Server

MitM captures shows that requests from members of H_a were detected and assigned *IP-MAC Mismatch* and *Red—Severe* as shown in Figure 22. Furthermore, some requests from members of H_m were flagged *Length Mismatch* and *Red—Severe* because they contained packets that had more than one Modbus frame (Figure 23). In Figure 24, our model flagged some requests from members of H_m as *Query Flooding of Known Read Query*, *EI: Below*

Threshold (E_i affected) and a *Blue—Guarded* in the query flooding captures. Investigations show that these were a result of delayed requests due to query flooding attacks from members of H_a. In the clean captures (Figure 25), there were two malicious requests from members of H_m that contained multiple Modbus frames; thus, they were flagged with *Length Mismatch* and *Red—Severe*.

Trust Score	Alert Level	Trust Level	Packet Count
-1.00	Red - Severe	Complete Distrust	5725

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: IP-MAC Mismatch | EF: Not Computed | EI: Not Computed | CONS: Not Computed	2862
SE: IP-MAC Mismatch-Modbus | EF: Not Computed | EI: Not Computed | CONS: Not Computed	2863

(**b**) Alert Details Summary.

Figure 22. Server: ATENA H2020 dataset external test-H_a in MitM capture.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	5144
0.99	Green - Low	Complete Trust	58,504
0.82	Green - Low	Very High Trust	7
-1.00	Red - Severe	Complete Distrust	11
0.83	Green - Low	Very High Trust	5

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	63,660
SE: Not Computed | EF: Length Mismatch | EI: Not Computed | CONS: Not Computed	11

(**b**) Alert Details Summary.

Figure 23. Server: ATENA H2020 dataset external test-H_m in MitM capture.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	342
0.99	Green - Low	Complete Trust	8223
0.56	Blue - Guarded	High Trust	177
0.46	Blue - Guarded	High Medium Trust	184
-1.00	Red - Severe	Complete Distrust	1
0.84	Green - Low	Complete Trust	1

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	8566
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Normal	177
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding of Known Read Query	184
SE: Not Computed | EF: Length Mismatch | EI: Not Computed | CONS: Not Computed	1

(**b**) Alert Details Summary.

Figure 24. Server: ATENA H2020 dataset external test-H_m in query flooding capture.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	994
0.99	Green - Low	Complete Trust	10,495
-1.00	Red - Severe	Complete Distrust	2
0.84	Green - Low	Complete Trust	1
0.83	Green - Low	Very High Trust	1

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	11,491
SE: Not Computed | EF: Length Mismatch | EI: Not Computed | CONS: Not Computed	2

(**b**) Alert Details Summary.

Figure 25. Server: ATENA H2020 dataset external test-H_M in clean capture.

13.2.2. Internal Attack Test towards Server

In MitM captures, Figure 26 shows that a member of H_a performed a baseline replay but did not perform any final attack and as such, was not detected by the model. However, when a baseline replay was performed and a final attack was done, it was detected and it showed the packet and multiple Modbus frames (Figure 27).

For query flooding captures, there are unknown writing requests that were flagged *Unknown Write Query* and *Red—Severe* as shown as Figure 28. Figure 29 shows Q sent by a member of H_a within less than time periods that is finally flagged with *Query Flooding Attack* and *Red—Severe* exceeding ζ_{pt}^{T}. For the sake of simplicity, we set ζ_{pt}^{T} to five requests even though it will vary from substation to substation. The first four requests were marked as *Blue—Guarded* and the fifth was marked as *Orange—High*. Figure 30 shows requests being flagged with *Red—Severe* because the packet manipulation attack by H_a triggered ϕ.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	13
0.99	Green - Low	Complete Trust	179

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	192

(**b**) Alert Details Summary.

Figure 26. ATENA H2020 dataset internal test-H_a in MitM capture-baseline replay.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	187
0.99	Green - Low	Complete Trust	2674
-1.00	Red - Severe	Complete Distrust	1
0.96	Green - Low	Complete Trust	1

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	2862
SE: Not Computed | EF: Length Mismatch | EI: Not Computed | CONS: Not Computed	1

(**b**) Alert Details Summary.

Figure 27. ATENA H2020 dataset internal test-H_a in MitM capture-baseline replay to final strike.

Trust Score	Alert Level	Trust Level	Packet Count
-1.00	Red - Severe	Complete Distrust	51,653

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Below Threshold | EF: Below Threshold | EI: Normal | CONS: Unknown Write Query Attack	51,653

(**b**) Alert Details Summary.

Figure 28. ATENA H2020 dataset internal test-H_a in query flooding capture-unknown write attack.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	1
0.56	Blue - Guarded	High Trust	1
0.46	Blue - Guarded	High Medium Trust	3
-0.43	Orange - High	High Medium Distrust	1
-1.00	Red - Severe	Complete Distrust	3408

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	1
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Normal	1
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding of Known Read Query	3
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding Attack	1
SE: Normal | EF: Normal | EI: Query Flooding | CONS: Query Flooding Attack	3408

(**b**) Alert Details Summary.

Figure 29. ATENA H2020 dataset internal test-H_a in query flooding capture-query flooding of known query.

Trust Score	Alert Level	Trust Level	Packet Count
-1.00	Red - Severe	Complete Distrust	13

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Not Computed | EF: Length Mismatch | EI: Not Computed | CONS: Not Computed	3
SE: Not Computed | EF: Not Computed | EI: Not Computed | CONS: Length < 12 Bytes	6
SE: Not Computed | EF: Not Computed | EI: Not Computed | CONS: Length > 260 Bytes	4

(**b**) Alert Details Summary.

Figure 30. ATENA H2020 dataset internal test-H_a in ping flood capture-maliciously crafted packets.

13.2.3. Internal Attack Test with IP-MAC Blacklisting towards Server

Figures starting from Figures 31–33 show that the packets that are dropped after β are less than β^T and the device is placed in H_b. Like the EPM dataset, it provides clarity on the attack caused by the violation. It also reveals that a compromised or attack device can be well behaved before acting to impair the target device.

13.2.4. Internal Attack Test towards Client

In this test, there were no external devices posing as servers; thus, we performed the internal test only. Attacks towards the client side from a member of H_s were mostly

affecting E_i. Further probing of the packets revealed that there were delayed responses to requests (Figures 34 and 35). We also observed that there replayed responses which saw a high increase time in query-response time. There were a few instances wherein the Modbus frame size exceeded the maximum frame size (Figure 36).

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	100
0.99	Green - Low	Complete Trust	1496
-1.00	Red - Severe	Complete Distrust	2533

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	1596
SE: Not Computed | EF: Length Mismatch | EI: Not Computed | CONS: Not Computed	1
SE: Not Computed | EF: Not Computed | EI: Not Computed | CONS: Blacklisted IP-MAC	2532

(**b**) Alert Details Summary.

Figure 31. Server: ATENA H2020 dataset internal test with IP-MAC blacklisting-H_a in MitM capture-baseline replay to final strike.

Trust Score	Alert Level	Trust Level	Packet Count
-1.00	Red - Severe	Complete Distrust	335,017

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Below Threshold | EF: Below Threshold | EI: Normal | CONS: Unknown Write Query Attack	1
SE: Not Computed | EF: Not Computed | EI: Not Computed | CONS: Blacklisted IP-MAC	335,016

(**b**) Alert Details Summary.

Figure 32. Server: ATENA H2020 dataset internal test with IP-MAC blocking-H_a in query flooding capture-unknown write attack.

Trust Score	Alert Level	Trust Level	Packet Count
1.00	Green - Low	Complete Trust	1
0.56	Blue - Guarded	High Trust	1
0.46	Blue - Guarded	High Medium Trust	3
-0.43	Orange - High	High Medium Distrust	1
-1.00	Red - Severe	Complete Distrust	22,008

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	1
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Normal	1
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding of Known Read Query	3
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding Attack	1
SE: Not Computed | EF: Not Computed | EI: Not Computed | CONS: Blacklisted IP-MAC	22,008

(**b**) Alert Details Summary.

Figure 33. Server: ATENA H2020 dataset internal test with IP-MAC blocking-H_a in query flooding capture-query flooding of known query.

Trust Score	Alert Level	Trust Level	Packet Count
0.99	Green - Low	Complete Trust	5277
0.89	Green - Low	Complete Trust	272
0.01	Yellow - Elevated	Low Trust	2
-0.08	Yellow - Elevated	Low Distrust	4
-0.98	Red - Severe	Complete Distrust	1
-1.00	Red - Severe	Complete Distrust	5

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	5277
SE: Normal | EF: Normal | EI: Normal | CONS: Query Flooding of Known Read Query	272
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Normal	2
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding of Known Read Query	4
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding Attack	1
SE: Normal | EF: Normal | EI: Query Flooding | CONS: Query Flooding Attack	5

(**b**) Alert Details Summary.

Figure 34. Client: ATENA H2020 dataset internal test-H_s in MitM capture.

Trust Score	Alert Level	Trust Level	Packet Count
0.99	Green - Low	Complete Trust	9865
0.89	Green - Low	Complete Trust	640
0.72	Green - Low	Very High Trust	34
0.71	Green - Low	Very High Trust	2
0.98	Green - Low	Complete Trust	1
0.01	Yellow - Elevated	Low Trust	4
-0.08	Yellow - Elevated	Low Distrust	3
0.00	Yellow - Elevated	Low Distrust	1

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	9902
SE: Normal | EF: Normal | EI: Normal | CONS: Query Flooding of Known Read Query	640
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Normal	4
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding of Known Read Query	3
SE: Normal | EF: Normal | EI: Normal | CONS: Query Flooding Attack	1

(**b**) Alert Details Summary.

Figure 35. Client: ATENA H2020 dataset internal test-H_s in query flooding capture.

Trust Score	Alert Level	Trust Level	Packet Count
0.99	Green - Low	Complete Trust	59,044
0.89	Green - Low	Complete Trust	3474
0.01	Yellow - Elevated	Low Trust	16
-0.08	Yellow - Elevated	Low Distrust	49
-0.98	Red - Severe	Complete Distrust	12
-1.00	Red - Severe	Complete Distrust	49
0.00	Yellow - Elevated	Low Distrust	6
0.82	Green - Low	Very High Trust	1

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	59,045
SE: Normal | EF: Normal | EI: Normal | CONS: Query Flooding of Known Read Query	3474
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Normal	16
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding of Known Read Query	49
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding Attack	12
SE: Normal | EF: Normal | EI: Query Flooding | CONS: Query Flooding Attack	48
SE: Normal | EF: Normal | EI: Normal | CONS: Query Flooding Attack	6
SE: Not Computed | EF: Not Computed | EI: Not Computed | CONS: Length > 260 Bytes	1

(**b**) Alert Details Summary.

Figure 36. Client: ATENA H2020 dataset internal test-H_s in ping flood DDOS capture.

13.2.5. Testing with Criticality Variation

The results presented to date had ϱ_i in reflecting a low criticality-ranked device in Table 9. Since most critical ranked devices provide $\varrho_i = 1$, those will generate a significant number of false alarms, and we use the weights specified in Section 11.5 to adjust to a suitable value. We implemented this on the client side to raise the necessary alarms for the critical IED. It can be observed that the results are more sensitive and this can be used to promptly raise alarms for critical devices for action to be taken on them. Comparing the Figures 34 and 37, it can be seen that Figure 37 is more sensitive.

13.3. Discussion

The results from our work showed that it was possible to characterize the attack of the datasets. Tests on the server side of the EPM dataset showed that E_s and E_f were the most affected because the attacks were more focused on TCP ports and Modbus read-only queries (see Figures 10–17). However, tests on the client side show that only E_s was affected (see Figures 18–21). The labelling of this dataset allowed us to determine the accuracy of our model as shown in Tables 11 and 12. Such confidence allows us to boldly claim similar accuracy with the ATENA H2020 dataset even though that dataset is not labelled. On the server side test, we observed that E_i, E_s, and E_f were affected by the attacks which shows how comprehensive the attacks were (see Figures 22–33). However, on the client side, it was mostly E_i that was affected by the attacks (see Figures 34–36).

Trust Score	Alert Level	Trust Level	Packet Count
0.99	Green - Low	Complete Trust	5277
0.39	Blue - Guarded	High Medium Trust	272
0.01	Yellow - Elevated	Low Trust	2
-0.58	Orange - High	High Distrust	4
-0.98	Red - Severe	Complete Distrust	1
-1.00	Red - Severe	Complete Distrust	5

(**a**) Alert/Trust Score Summary.

Alert Description	Packet Count
SE: Normal | EF: Normal | EI: Normal | CONS: Normal	5277
SE: Normal | EF: Normal | EI: Normal | CONS: Query Flooding of Known Read Query	272
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Normal	2
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding of Known Read Query	4
SE: Normal | EF: Normal | EI: Below Threshold | CONS: Query Flooding Attack	1
SE: Normal | EF: Normal | EI: Query Flooding | CONS: Query Flooding Attack	5

(**b**) Alert Details Summary.

Figure 37. Client: ATENA H2020 dataset internal test with higher criticality weight.

Furthermore, we identified that a description of the mix of attacks was not referenced by the authors. We noticed that different works—not on trust, however—do not give specifics as our work has done in the wake of identifying attacks [112–114]. Furthermore, the comparison of our work with other trust models was challenging because there was only one trust model [84] that was utilized in a scenario such as ours. Notwithstanding, our trust model which computed trust made on the ratio of responses to requests and that would fail against baseline replay attacks. Their model would also not detect attacks contained in responses.

We noticed that the ATENA H2020 dataset had the same transaction ID throughout, and such an implementation makes it easy for an attacker to include malicious packets because there are no similarities in the MBAP header. We recommend that transaction IDs are made sequential to enable the tracking of packets. We also recommend that each request must utilize one session per request to mitigate TCP session hijackings. We also recommend that stacked Modbus PDU requests be dropped by an application's Modbus implementation.

14. Conclusions

In this paper, we present a categorized review of literature related to trust within the Smart Grid. This categorization was guided by the trust definitions according to the literature and the NIST priority areas and conceptual domains. From the presented paper, it is very clear that a lot of work needs to be done in the field of trust within the Smart Grid as well as making efforts to have it implemented in a cognitive environment whereby components can be adaptable to situations.

We also presented and tested a novel trust model for substations that detects attacks within the substation. We stated that familiarity and consequence are required to compute trust. We included the output of the novel risk assessment tool to compute the consequence of an attack on a substation. Using the model, we tested our work on two publicly available datasets using three kinds of tests. The external test is one in which purely attacker devices (not compromised substation devices) are assumed to be not part of the substation's network. The second is the internal test wherein all devices are assumed to be part of the substation's network. The final test is the internal test with the IP-MAC

block which assumes the position of the second test but blacklists any device that sends a malicious message.

Our model also revealed the behaviour of the datasets which has not been done in other trust models and not detailed as such in papers that used those datasets.

15. Future Work

We believe that our model can be embedded in a device's logic, extended to other OT-based protocols such as DNP3 (future work), and implemented in other critical infrastructures. Queries made out of order during troubleshooting will create false alarms; thus, this is a weakness of our work and will be addressed in future work. We aim to look at the community computation of trust for future work for multiple devices to manage trust-based attacks. The trust transferability of a device from one substation to another is also marked for future work. We also observed that a Modbus dataset containing network captures and attack scenarios specific to substations is required and that will be addressed in future work.

Author Contributions: Conceptualization and methodology, K.B.B., A.A.G., and A.H.L; software, K.B.B; validation, formal analysis and investigation, K.B.B., A.A.G., and A.H.L; writing—original draft preparation, K.B.B.; writing—review, editing and visualization K.B.B., A.A.G., and A.H.L; supervision, A.A.G. and A.H.L; funding acquisition, A.A.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Atlantic Canada Opportunities Agency (ACOA) through the Atlantic Innovation Fund (AIF) project #212420 and the Natural Sciences and Engineering Research Council of Canada—NSERC (Grant# RGPIN 231074).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors acknowledge the funding from the Atlantic Canada Opportunities Agency (ACOA) through the Atlantic Innovation Fund (AIF) project #212420 and a grant from the Natural Sciences and Engineering Research Council of Canada—NSERC (Grant# RGPIN 231074)—to Ali Ghorbani.

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

Abbreviations

The following abbreviations are used in this manuscript:

ACM	Association for Computing Machinery
AMI	Advanced Metering Infrastructure
APT	Advanced Persistent Threat
ATENA	Advanced Tools to Assess and Mitigate the Criticality of ICT Components and their Dependencies over Critical Infrastructures
CB	Circuit Breaker
CHP	Combined Heat and Power
CnC	Command and Control
CNP	Contract Net Protocol
CONS	Consequence
CT	Current Transformer
CVE	Common Vulnerabilities and Exposures
CWE	Common Weakness Enumeration
DER	Distributed Energy Resources
DL	Direct Line
DOS	Denial of Service
EF	Exposure Frequency
EI	Exposure Intensity
EPM	École Polytechnique de Montréal

FIPA	Foundation for Intelligent Physical Agents
GOOSE	Generic Object-Oriented Substation Event
HMI	Human–Machine Interface
IBM	International Business Machines Company
IDS	Intrusion Detection Systems
IED	Intelligent Electronic Device
IEEE	Institute of Electrical and Electronics Engineers
IL	Incoming Line
IP	Internet Protocol
IP	Internet Protocol
JADE	Java Agent Development Framework
KQML	Knowledge Query and Manipulation Language
LMP	Local Marginal Price
MAC	Media Access Control
MAS	Multi-Agent System
MiTM	Man in the Middle
MS-ISAC	Multi-State Information Sharing and Analysis Center
NIST	National Institute of Standards and Technology
OC	Outgoing Circuit
OL	Outgoing Line
OT	Operational Technology
PEV	Plug-in Electric Vehicles
PNNL	Pacific Northwest National Laboratory
PT	Potential Transformer
SAS	Substation Automation System
SCADA	Supervisory Control and Data Acquisition
SE	Similar Exposure
SPS	Special Protection System
TCP	Transmission Control Protocol
TTL	Time to Live
TX	Transformer
WASA	Wide-Area Situational Awareness
WCA	Water Cycle Algorithm
WSN	Wireless Sensor Network

Appendix A

Table A1. Table of notations.

Notation	Meaning
a_i	Agent
a_j	Subject
r_{ij}	Risk between a_i and a_j
α_{ij}	A transaction between a_i and a_j
k_{ij}^t	Knowledge about α_{ij}
k_{ij}'	Knowledge of previous transactions between a_i and a_j
k_{a_j}	Knowledge of a_j
t	Time
T_{ij}	Trust between a_i and a_j
T_{ij}'	Previous trust between a_i and a_j
d_i	Agent device
d_j	Subject device

Table A1. *Cont.*

Notation	Meaning
m_{ij}	Message between d_i and d_j
h_{ij}	History of communication between d_i and d_j
D	A list of n devices
R_{d_i}	List of devices functionally dependent on d_i
An_{d_i}	List of devices that functionally influence d_I
I_{d_i}	Intersection of R_{d_i} and An_{d_i}
l	Criticality rank of devices
Ξ	Substation
M	A set of clients
S	A set of servers
N	A set of network devices
Q	A set of queries
R	A set of responses associated with Q
ϑ	ype of query or response being either read or write
Q'	A malicious Q
E_i	Exposure intensity
E_f	Exposure frequency
E_s	Similar exposure
E^T	An exposure's threshold
κ_E	An alarm associated with a particular exposure factor of familiarity
Z	A set of features associated with E_i
Z_R	A reference set of features associated with E_i
ζ_{pt}	Pre-time feature
ζ_{qq}	Inter-query time feature
ζ_{rr}	Inter-response time feature
ζ_{qr}	Query-response time feature
ζ_{tt}	Transaction time feature
ζ_{to}	Timeout feature
ζ_{qq}^T	Inter-query time threshold
ζ_{rr}^T	Inter-response time threshold
ζ_{qr}^T	Query-response time threshold
ζ_{to}^T	Timeout threshold
Y	Moore machine used to generate E_s-based features
ρ	Finite set of states
ρ_{rdi}	Read discrete input state
ρ_{rc}	Read coil state
ρ_{wsc}	Write coil state
ρ_{wmc}	Write multiple coils state
ρ_{rhr}	Read holding registers state
ρ_{wsr}	Write single register state
ρ_{wmr}	Write multiple registers state
ρ_{rir}	Read input registers state
ρ_u	Unknown state
fc_i	Modbus function code of q_i or r_i

Table A1. *Cont.*

Notation	Meaning
a	Modbus address
d_{q_i}	Modbus data value of a
b_{q_i}	Modbus byte count of the value found at a
l_d	Modbus length of data frame
l_f	Length of entire Modbus packet
ι_{q_i}	Modbus coil/discrete input/input register/holding register quantity
$l_{h_{q_i}}$	Modbus header length
σ	A set of input alphabets of Y
δ	A transition function of Y
Ψ	A set of features associated with E_s
Ψ_R	A reference set of features associated with E_s
ψ_s	State traversed feature
ψ_η	IP-MAC mismatch feature
ψ_p	Port mismatch feature
ψ_{us}	Unknown state feature
ψ_{ma}	Address match feature
ψ_{mas}	Address size match feature
ψ_{fc}	Function code match feature
ψ_{mdir}	Discrete input reference match feature
ψ_{mdiq}	Discrete input quantity match feature
ψ_{mcr}	Coil reference match feature
ψ_{mcq}	Coil quantity match feature
ψ_{mhrr}	Holding register reference match feature
ψ_{mhrq}	Holding register quantity feature
ψ_{mirr}	Input register reference match
ψ_{mirq}	Input register quantity match
λ	Output function of Y
Γ	A set of features associated E_f
Γ_R	A reference set of features associated E_f
γ_{frc}	Count for read coil function code
γ_{cq}	Coil quantity
γ_{frdi}	Count for read discrete input function code
γ_{diq}	Discrete input quantity
γ_{frhr}	Count for read holding register function code
γ_{hrq}	Holding register quantity
γ_{frir}	Count for read input register function code
γ_{irq}	Input register quantity
γ_{fwsc}	Count for write single coil function code
γ_{cv}	Coil value
γ_{cdc}	Coil data byte count
γ_{didc}	Discrete input data byte count
γ_{hrdc}	Holding register data byte count
γ_{irdc}	Input register data byte count
γ_{fwsr}	Count for write single register function code

Table A1. *Cont.*

Notation	Meaning
γ_{hrv}	Holding register value
γ_{fwmc}	Count for write multiple coils function code
γ_{cvs}	Set of coil values
γ_{fwmr}	Count for Write Multiple Registers function code
γ_{hrvs}	Set of holding register values
γ_{irv}	Input register value
γ_{irvs}	Set of input register values
γ_{fs}	Frame size feature
F	Familiarity
φ_{ω}	Replay sensitivity weight
φ'_{ω}	Replay sensitivity weight for unknown states
φ_{ξ}	Reconnaissance sensitivity weight
φ_{χ}	Query flooding sensitivity weight
φ'_{χ}	Query flooding sensitivity weight for unknown states
ϱ	Criticality rank ratio
τ	Environment status attack value
ω	Replay attack value
ξ	Reconnaissance attack value
χ	Query flooding attack value
ϕ	Packet manipulation attack value
C	Consequence
β	Trust score
θ_I	Initial state of device
β_i^o	Previous trust score
β^T	Trust score threshold
μ	Forgiveness weight
θ_{μ}	Forgiveness state of device
H_m	Client group
H_s	Server group
H_a	Attack group
H_t	Targeted group
H'	Compromised group
H'_m	Compromised client group
H_b	Blacklisted group

References

1. Gupta, B.; Akhtar, T. A survey on smart power grid: Frameworks, tools, security issues, and solutions. *Ann. Telecommun.* **2017**, *72*, 517–549. [CrossRef]
2. Knirsch, F.; Unterweger, A.; Unterrainer, M.; Engel, D. Comparison of the Paillier and ElGamal Cryptosystems for Smart Grid Aggregation Protocols. In Proceedings of the ICISSP, Valletta, Malta, 25–27 February 2020; pp. 232–239.
3. Abbasinezhad-Mood, D.; Nikooghadam, M. Design and extensive hardware performance analysis of an efficient pairwise key generation scheme for smart grid. *Int. J. Commun. Syst.* **2018**, *31*, e3507. [CrossRef]
4. Yu, S.; Park, K.; Lee, J.; Park, Y.; Park, Y.; Lee, S.; Chung, B. Privacy-preserving lightweight authentication protocol for demand response management in smart grid environment. *Appl. Sci.* **2020**, *10*, 1758. [CrossRef]
5. Gunduz, M.Z.; Das, R. Cyber-security on smart grid: Threats and potential solutions. *Comput. Netw.* **2020**, *169*, 107094. [CrossRef]
6. Langner, R. Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Secur. Priv.* **2011**, *9*, 49–51. [CrossRef]

7. Greer, C.; Wollman, D.; Prochaska, D.; Boynton, P.; Mazer, J.; Nguyen, C.; FitzPatrick, G.; Nelson, T.; Koepke., G; Hefner, A., Jr.; et al. NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 3.0. Available online: https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=916755 (accessed on 21 October 2020).
8. Sommerville, P.B. *Distributed Energy Resources: The Role of Regional Planning, New Benefit-Cost Methodologies and the Competitive Landscape*; Mowat Centre for Policy Innovation: Toronto, ON, Canada, 2019.
9. Kent Hedrick. What New Capabilities Mean for Distribution Grid Management. Available online: https://www.landisgyr.com/webfoo/wp-content/uploads/2014/07/LAN-14009_GridMgmtWP_140728.pdf (accessed on 17 November 2020).
10. Xia, J.; Wang, Y. Secure key distribution for the smart grid. *IEEE Trans. Smart Grid* **2012**, *3*, 1437–1443. [CrossRef]
11. Bartoli, A.; Hernandez-Serrano, J.; Soriano, M.; Dohler, M.; Kountouris, A.; Barthel, D. Secure lossless aggregation for smart grid M2M networks. In Proceedings of the 2010 First IEEE International Conference on Smart Grid Communications, Gaithersburg, MD, USA, 4–6 October 2010; pp. 333–338.
12. Maharjan, S.; Zhu, Q.; Zhang, Y.; Gjessing, S.; Basar, T. Dependable demand response management in the smart grid: A Stackelberg game approach. *IEEE Trans. Smart Grid* **2013**, *4*, 120–132. [CrossRef]
13. Parvania, M.; Fotuhi-Firuzabad, M. Demand response scheduling by stochastic SCUC. *IEEE Trans. Smart Grid* **2010**, *1*, 89–98. [CrossRef]
14. Deilami, S.; Masoum, A.S.; Moses, P.S.; Masoum, M.A. Real-time coordination of plug-in electric vehicle charging in smart grids to minimize power losses and improve voltage profile. *IEEE Trans. Smart Grid* **2011**, *2*, 456–467. [CrossRef]
15. Carryl, C.; Ilyas, M.; Mahgoub, I.; Rathod, M. The PEV security challenges to the smart grid: Analysis of threats and mitigation strategies. In Proceedings of the 2013 International Conference on Connected Vehicles and Expo (ICCVE), Las Vegas, NV, USA, 2–6 December 2013, pp. 300–305.
16. Liu, J.; Xiao, Y.; Li, S.; Liang, W.; Chen, C.P. Cyber security and privacy issues in smart grids. *IEEE Commun. Surv. Tutor.* **2012**, *14*, 981–997. [CrossRef]
17. Liu, N.; Chen, J.; Zhu, L.; Zhang, J.; He, Y. A key management scheme for secure communications of advanced metering infrastructure in smart grid. *IEEE Trans. Ind. Electron.* **2012**, *60*, 4746–4756. [CrossRef]
18. Qi, J.; Hahn, A.; Lu, X.; Wang, J.; Liu, C.C. Cybersecurity for distributed energy resources and smart inverters. *IET Cyber-Phys. Syst. Theory Appl.* **2016**, *1*, 28–39. [CrossRef]
19. Srikantha, P.; Kundur, D. A DER attack-mitigation differential game for smart grid security analysis. *IEEE Trans. Smart Grid* **2015**, *7*, 1476–1485. [CrossRef]
20. Lu, R.; Liang, X.; Li, X.; Lin, X.; Shen, X. EPPA: An efficient and privacy-preserving aggregation scheme for secure smart grid communications. *IEEE Trans. Parallel Distrib. Syst.* **2012**, *23*, 1621–1631.
21. Radoglou–Grammatikis, P.I.; Sarigiannidis, P.G. Securing the smart grid: A comprehensive compilation of intrusion detection and prevention systems. *IEEE Access* **2019**, *7*, 46595–46620. [CrossRef]
22. Goodman, S. Alliance Commission on National Energy Efficiency Policy. Available online: https://www.ase.org/history-energy-efficiency-alliance-commission-national-energy-efficiency-policy (accessed on 17 May 2020)
23. Cook, K. Trust in Society. In *Russell Sage Foundation Series on Trust, New York*; Publisher is Russell Sage: New York, NY, USA, 2003; Volume 2, p. 432.
24. Gambetta, D. Can we trust trust. *Trust. Mak. Break. Coop. Relat.* **2000**, *13*, 213–237.
25. Marsh, S.P. Formalising trust as a computational concept. In *STORRE: Stirling Online Research Repository*; University of Stirling: Stirling, UK, 1994.
26. Joint Task Force Transformation Initiative. *SP 800-39. Managing Information Security Risk: Organization, Mission, and Information System View*; NIST: Scotts Valley, CA, USA, 2011.
27. Altaf, A.; Abbas, H.; Iqbal, F.; Derhab, A. Trust models of internet of smart things: A survey, open issues, and future directions. *J. Netw. Comput. Appl.* **2019**, *137*, 93–111. [CrossRef]
28. Fung, C.J.; Zhang, J.; Aib, I.; Boutaba, R. Dirichlet-based trust management for effective collaborative intrusion detection networks. *IEEE Trans. Netw. Serv. Manag.* **2011**, *8*, 79–91. [CrossRef]
29. Cheng, X.; Li, T. A credibility measurement method of smart grid data. In Proceedings of the 2016 IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Xi'an, China, 3–5 October 2016; pp. 1231–1235.
30. Velusamy, D.; Pugalendhi, G.; Ramasamy, K. A Cross-Layer Trust Evaluation Protocol for Secured Routing in Communication Network of Smart Grid. *IEEE J. Sel. Areas Commun.* **2019**, *38*, 193–204. [CrossRef]
31. Velusamy, D.; Pugalendhi, G. Water Cycle Algorithm Tuned Fuzzy Expert System for Trusted Routing in Smart Grid Communication Network. *IEEE Trans. Fuzzy Syst.* **2020**, *28*, 1167–1177. [CrossRef]
32. Alnasser, A.; Sun, H. A fuzzy logic trust model for secure routing in smart grid networks. *IEEE Access* **2017**, *5*, 17896–17903. [CrossRef]
33. Xiang, M.; Bai, Q.; Liu, W. Self-adjustable trust-based energy efficient routing for smart grid systems. In Proceedings of the 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, Macau, China, 4–7 December 2012; Volume 3, pp. 378–382.

34. Xiang, M.; Liu, W.; Bai, Q. Trust-based geographical routing for smart grid communication networks. In Proceedings of the 2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm), Tainan, Taiwan, 5–8 November 2012; pp. 704–709.
35. Xiang, M.; Liu, W.; Bai, Q. TIGER: A trust-based intelligent geographical energy-aware routing for smart grid communication networks. In Proceedings of the 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, Melbourne, Australia, 16–18 July 2013; pp. 65–72.
36. Bello, A.; Liu, W.; Bai, Q.; Narayanan, A. Revealing the role of topological transitivity in efficient trust and reputation system in smart metering network. In Proceedings of the 2015 IEEE International Conference on Data Science and Data Intensive Systems, Sydney, Australia, 11–13 December 2015; pp. 337–342.
37. Bello, A.; Liu, W.; Bai, Q.; Narayanan, A. Exploring the role of structural similarity in securing smart metering infrastructure. In Proceedings of the 2015 IEEE International Conference on Data Science and Data Intensive Systems, Sydney, Australia, 11–13 December 2015; pp. 343–349.
38. Reza, S.S.; Mahbub, T.N.; Islam, M.M.; Arifeen, M.M.; Remu, S.R.H.; Hossain, D.A. Assuring Cyber Security in Smart Grid Networks by Fuzzy-logic based Trust Management Model. In Proceedings of the 2019 IEEE International Conference on Robotics, Automation, Artificial-intelligence and Internet-of-Things (RAAICON), Dhaka, Bangladesh, 29 November–1 December 2019; pp. 1–4.
39. Pliatsios, D.; Sarigiannidis, P.; Efstathopoulos, G.; Sarigiannidis, A.; Tsiakalos, A. Trust Management in Smart Grid: A Markov Trust Model. In Proceedings of the 2020 9th International Conference on Modern Circuits and Systems Technologies (MOCAST), Bremen, Germany, 7–9 September 2020; pp. 1–4.
40. Pradhan, O.; Awan, M.; Newman, K.; Barnes, F. Trust and reputation approach to smart grid security. In Proceedings of the 2011 4th IEEE International Symposium on Resilient Control Systems, Boise, ID, USA, 9–11 August 2011; pp. 101–104.
41. Fadul, J.; Hopkinson, K.; Andel, T.; Kurkowski, S.; Moore, J. Simple trust protocol for wired and wireless SCADA networks. In Proceedings of the International Conference on Cyber Warfare and Security, Dayton, OH, USA, 8–9 April 2010; Academic Conferences International Limited: Reading, UK 2010; p. 89.
42. Fadul, J.E.; Hopkinson, K.M.; Andel, T.R.; Sheffield, C.A. A trust-management toolkit for smart-grid protection systems. *IEEE Trans. Power Deliv.* **2013**, *29*, 1768–1779. [CrossRef]
43. Shipman, C.M.; Hopkinson, K.M.; Lopez, J. Con-resistant trust for improved reliability in a smart-grid special protection system. *IEEE Trans. Power Deliv.* **2014**, *30*, 455–462. [CrossRef]
44. Zhang, Y.; Sun, W.; Wang, L. Location and communication routing optimization of trust nodes in smart grid network infrastructure. In Proceedings of the 2012 IEEE Power and Energy Society General Meeting, San Diego, CA, USA, 22–26 July 2012; pp. 1–8.
45. Zhang, Y.; Sun, W.; Wang, L. Efficient trust node aware routing in ZigBee communication network of smart grid. In Proceedings of the 2012 10th International Power & Energy Conference (IPEC), Ho Chi Minh City, Vietnam, 12–14 December 2012; pp. 321–326.
46. Zhang, Y.; Sun, W.; Wang, L. Placement of primary-secondary trust nodes in smart grid communication network. In Proceedings of the 2013 IEEE Power & Energy Society General Meeting, Vancouver, BC, Canada, 21–25 July 2013; pp. 1–5.
47. Zhang, Y.; Sun, W.; Wang, L. Fault-tolerant optimal routing of trust nodes in smart grid communications. In Proceedings of the 2012 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), Bangkok, Thailand, 27–31 May 2012; pp. 281–286.
48. Zhang, Y.; Wang, L.; Sun, W. Trust system design optimization in smart grid network infrastructure. *IEEE Trans. Smart Grid* **2013**, *4*, 184–195. [CrossRef]
49. Hasan, M.M.; Mouftah, H.T. Optimization of trust node assignment for securing routes in smart grid SCADA networks. *IEEE Syst. J.* **2018**, *13*, 1505–1513. [CrossRef]
50. McDonald, J.D. *Electric Power Substations Engineering*; CRC Press: Boca Raton, FL, USA, 2016.
51. Hauser, C.H.; Bakken, D.E.; Dionysiou, I.; Gjermundrod, K.H.; Irava, V.; Helkey, J.; Bose, A. Security, trust, and QoS in next-generation control and communication for large power systems. *Int. J. Crit. Infrastruct.* **2008**, *4*, 3–16. [CrossRef]
52. Singh, N.K.; Mahajan, V. Cyber Attack Detection In Smart Grid Substation Using Virtual Range Increment Furthermore, Trust Weight. In Proceedings of the 2019 8th International Conference on Power Systems (ICPS), Jaipur, India, 20–22 December 2019; pp. 1–6.
53. Singh, N.K.; Mahajan, V. Detection of cyber cascade failure in smart grid substation using advance grey wolf optimization. *J. Interdiscip. Math.* **2020**, *23*, 69–79. [CrossRef]
54. Obert, J.; Chavez, A.; Johnson, J. Behavioral based trust metrics and the smart grid. In Proceedings of the 2018 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE), New York, NY, USA, 1–3 August 2018; pp. 1490–1493.
55. Obert, J.; Chavez, A. Graph-Based Event Classification in Grid Security Gateways. In Proceedings of the 2019 Second International Conference on Artificial Intelligence for Industries (AI4I), Laguna Hills, CA, USA, 25–27 September 2019; pp. 63–66.
56. Nasr, P.M.; Yazdian-Varjani, A. Toward operator access management in SCADA system: Deontological threat mitigation. *IEEE Trans. Ind. Inform.* **2017**, *14*, 3314–3324. [CrossRef]
57. Rashid, M.T.A.; Yussof, S.; Yusoff, Y. Trust system architecture for securing GOOSE communication in IEC 61850 substation network. *Int. J. Secur. Its Appl.* **2016**, *10*, 289–302. [CrossRef]

58. Bellifemine, F.L.; Caire, G.; Greenwood, D. *Developing Multi-Agent Systems with JADE*; John Wiley & Sons: Hoboken, NJ, USA, 2007; Volume 7.
59. Kuzin, A.Y.; Demidova, G.L.; Lukichev, D.V. An Approach of the JADE and Simulink Interaction to Control Smart Grid Based on the Multi Agent System. In Proceedings of the 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), Saint Petersburg and Moscow, Russia, 28–31 January 2019; pp. 574–577.
60. Maaroufi, M.; Ouassaid, M. Demand side management in smart grid by multi-agent systems technology. In Proceedings of the 2014 International Conference on Multimedia Computing and Systems (ICMCS), Marrakesh, Morocco, 14–16 April 2014; pp. 1042–1045.
61. Garrab, A.; Bouallegue, A.; Bouallegue, R. Multi-Agent modeling of a meters network used in Smart Grid. In Proceedings of the 2014 World Congress on Computer Applications and Information Systems (WCCAIS), Hammamet, Tunisia, 17–19 January 2014; pp. 1–5.
62. Kim, B.; Lavrova, O. Optimal power flow and energy-sharing among multi-agent smart buildings in the smart grid. In Proceedings of the 2013 IEEE Energytech, Cleveland, OH, USA, 21–23 May 2013; pp. 1–5.
63. Dong, L.; Li, Y.; Liu, K.; Pu, T.; Liu, G. Research on smart grid simulation framework based on distributed intelligent system. In Proceedings of the 2014 International Conference on Power System Technology, Chengdu, China, 20-22 October 2014; pp. 1969–1974.
64. JAVA Agent DEvelopment Framework. Available online: https://jade.tilab.com/ (accessed on 23 March 2020).
65. Kantamneni, A.; Brown, L.E.; Parker, G.; Weaver, W.W. Survey of multi-agent systems for microgrid control. *Eng. Appl. Artif. Intell.* **2015**, *45*, 192–203. [CrossRef]
66. Bellifemine, F.; Caire, G.; Poggi, A.; Rimassa, G. JADE: A software framework for developing multi-agent applications. Lessons learned. *Inf. Softw. Technol.* **2008**, *50*, 10–21. [CrossRef]
67. Volttron | Devices | Data | Decisions. Available online: https://volttron.org/ (accessed on 4 March 2021).
68. Welcome to the Aglets Web Site. Available online: http://aglets.sourceforge.net/ (accessed on 4 March 2021).
69. AOS Group: Products. Available online: https://aosgrp.com/products/jack/ (accessed on 2 March 2021).
70. Shakshuki, E.; Jun, Y. Multi-agent development toolkits: An evaluation. In Proceedings of the International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, Ottawa, ON, Canada, 17–20 May 2004; Springer: Berlin/Heidelberg, Germany, 2004; pp. 209–218.
71. Zhao, X.; Hu, G.; Wu, Z. The Smart grid scheduling based on contract net protocol with trust model. In Proceedings of the 2014 IEEE/ACIS 13th International Conference on Computer and Information Science (ICIS), Taiyuan, China, 4–6 June 2014; pp. 419–424.
72. Smith, R.G. The contract net protocol: High-level communication and control in a distributed problem solver. *IEEE Trans. Comput.* **1980**, *29*, 1104–1113. [CrossRef]
73. Alavikia, Z.; Mozayani, N.; Shahbazi, J.; Alavikia, F. Utilizing an Agent Based Negotiation Mechanism to Defend Against Jamming Attack in Smart Grid Power Market. In Proceedings of the 2018 9th International Symposium on Telecommunications (IST), Tehran, Iran, 17–19 December 2018; pp. 45–52.
74. Conejo, A.J.; Castillo, E.; Mínguez, R.; Milano, F. Locational marginal price sensitivities. *IEEE Trans. Power Syst.* **2005**, *20*, 2026–2033. [CrossRef]
75. Li, F.; Bo, R. Small test systems for power system economic studies. In Proceedings of the IEEE PES General Meeting, Minneapolis, Minnesota, USA, 25–29 July 2010; pp. 1–4.
76. Pereira, A.; Rodrigues, N.; Barbosa, J.; Leitão, P. Trust and risk management towards resilient large-scale cyber-physical systems. In Proceedings of the 2013 IEEE International Symposium on Industrial Electronics, Taipei, Taiwan, 28–31 May 2013; pp. 1–6.
77. Chassin, D.P.; Schneider, K.; Gerkensmeyer, C. GridLAB-D: An open-source power systems modeling and simulation environment. In Proceedings of the 2008 IEEE/PES Transmission and Distribution Conference and Exposition, Chicago, IL, USA, 21–24 April 2008; pp. 1–5.
78. Cintuglu, M.H.; Ishchenko, D. Secure Distributed State Estimation for Networked Microgrids. *IEEE Internet Things J.* **2019**, *6*, 8046–8055. [CrossRef]
79. Matei, I.; Baras, J.S.; Srinivasan, V. Trust-based multi-agent filtering for increased smart grid security. In Proceedings of the 2012 20th Mediterranean Conference on Control & Automation (MED), Barcelona, Spain, 3–6 July 2012; pp. 716–721.
80. Cunningham, C.; Roque, A. Adapting an agent-based model of socio-technical systems to analyze security failures. In Proceedings of the 2017 IEEE International Symposium on Technologies for Homeland Security (HST), Waltham, MA, USA, 25-26 April 2017; pp. 1–7.
81. Minkel, J. The 2003 Northeast Blackout–Five Years Later. *Sci. Am.* **2008**, *13*, 1–3.
82. Hussain, S.; Honeth, N.; Gustavsson, R.; Sandels, C.; Saleem, A. Trustworthy injection/curtailment of DER in distribution network maintaining quality of service. In Proceedings of the 2011 16th International Conference on Intelligent System Applications to Power Systems, Hersonissos, Greece, 25–28 September 2011; pp. 1–6.
83. AOS Group | Products. Available online: https://www.aosgrp.com/products/jack/ (accessed on 2 March 2021).
84. Borowski, J.F.; Hopkinson, K.M.; Humphries, J.W.; Borghetti, B.J. Reputation-based trust for a cooperative agent-based backup protection scheme. *IEEE Trans. Smart Grid* **2011**, *2*, 287–301. [CrossRef]

85. Chen, Q.; Schmidt-Eisenlohr, F.; Jiang, D.; Torrent-Moreno, M.; Delgrossi, L.; Hartenstein, H. Overhaul of IEEE 802.11 modeling and simulation in ns-2. In Proceedings of the 10th ACM Symposium on Modeling, Analysis, and Simulation of Wireless and Mobile Systems, Chania Crete Island Greece, 22–26 October, 2007; pp. 159–168.
86. Hopkinson, K.; Wang, X.; Giovanini, R.; Thorp, J.; Birman, K.; Coury, D. EPOCHS: A platform for agent-based electric power and communication simulation built from commercial off-the-shelf components. *IEEE Trans. Power Syst.* **2006**, *21*, 548–558. [CrossRef]
87. Anaya-Lara, O.; Acha, E. Modeling and analysis of custom power systems by PSCAD/EMTDC. *IEEE Trans. Power Deliv.* **2002**, *17*, 266–272. [CrossRef]
88. Blangenois, J.; Guemkam, G.; Feltus, C.; Khadraoui, D. Organizational security architecture for critical infrastructure. In Proceedings of the 2013 International Conference on Availability, Reliability and Security, Regensburg, Germany, 2–6 September 2013; pp. 316–323.
89. Guemkam, G.; Blangenois, J.; Feltus, C.; Khadraoui, D. Metamodel for reputation based agents system: Case study for electrical distribution SCADA design. In Proceedings of the 6th International Conference on Security of Information and Networks, Aksaray, Turkey, 26–28 November 2013; pp. 251–255.
90. Cho, J.H.; Chan, K.; Adali, S. A survey on trust modeling. *ACM Comput. Surv. (CSUR)* **2015**, *48*, 1–40. [CrossRef]
91. Yonelinas, A.P. The nature of recollection and familiarity: A review of 30 years of research. *J. Mem. Lang.* **2002**, *46*, 441–517. [CrossRef]
92. Zhan, C.; Li, W.; Ogunbona, P. Measuring the degree of face familiarity based on extended NMF. *ACM Trans. Appl. Percept. (TAP)* **2013**, *10*, 1–22. [CrossRef]
93. Tonkoski, R.; Lopes, L.A.C.; El-Fouly, T.H.M. Coordinated Active Power Curtailment of Grid Connected PV Inverters for Overvoltage Prevention. *IEEE Trans. Sustain. Energy* **2011**, *2*, 139–147. [CrossRef]
94. Boakye-Boateng, K.; Ghorbani, A.A.; Lashkari, A.H. RiskISM: A Risk Assessment Tool for Substations. In Proceedings of the 2021 IEEE 9th International Conference on Smart City and Informatization (iSCI), Shenyang, China, 20–22 October 2021; pp. 23–30. [CrossRef]
95. Single Line Diagrams Of Substations 66/11 kV and 11/0.4 kV: EEP. Available online: https://electrical-engineering-portal.com/single-line-diagrams-substations. (accessed on 12 January 2021).
96. Kim, B.K.; Kang, Y. Abnormal traffic detection mechanism for protecting IIoT environments. In Proceedings of the 2018 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, 17–19 October 2018; pp. 943–945.
97. Case, D.U. Analysis of the cyber attack on the Ukrainian power grid. *Electr. Inf. Shar. Anal. Cent. (E-ISAC)* **2016**, *388*, 1–29.
98. Nelson, N. The Impact of Dragonfly Malware on inDustrial Control Systems. SANS Institute: Rockville, MD, USA, 2016.
99. North American Protective Relay Marketplace. Available online: http://www.newton-evans.com/north-american-protective-relay-marketplace-new-report-now-available/ (accessed on 13 January 2021).
100. Modbus Organization. MODBUS Messaging on TCP/IP Implementation Guide: V1.0b. Available online: https://www.modbus.org/docs/Modbus_Messaging_Implementation_Guide_V1_0b.pdf (accessed on 23 June 2021).
101. Modbus Organization. Modbus Application Protocol Specification V1.1b. Available online: https://modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf (accessed on 25 June 2021).
102. Tidrea, A.; Korodi, A.; Silea, I. Cryptographic Considerations for Automation and SCADA Systems Using Trusted Platform Modules. *Sensors* **2019**, *19*, 4191. [PubMed]
103. Lai, Y.; Gao, H.; Liu, J. Vulnerability Mining Method for the Modbus TCP Using an Anti-Sample Fuzzer. *Sensors* **2020**, *20*, 2040. [CrossRef]
104. Siniosoglou, I.; Radoglou–Grammatikis, P.; Efstathopoulos, G.; Fouliras, P.; Sarigiannidis, P. A unified deep learning anomaly detection and classification approach for smart grid environments. *IEEE Trans. Netw. Serv. Manag.* **2021**, *18*, 1137–1151. [CrossRef]
105. Nyasore, O.N.; Zavarsky, P.; Swar, B.; Naiyeju, R.; Dabra, S. Deep packet inspection in industrial automation control system to mitigate attacks exploiting modbus/TCP vulnerabilities. In Proceedings of the 2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS), Baltimore, MD, USA, 25–27 May 2020; pp. 241–245.
106. Si, Y.; Korada, N.; Ayyanar, R.; Lei, Q. A high performance communication architecture for a smart micro-grid testbed using customized edge intelligent devices (eids) with spi and modbus tcp/ip communication protocols. *IEEE Open J. Power Electron.* **2021**, *2*, 2–17. [CrossRef]
107. González, I.; Calderón, A.J.; Portalo, J.M. Innovative Multi-Layered Architecture for Heterogeneous Automation and Monitoring Systems: Application Case of a Photovoltaic Smart Microgrid. *Sustainability* **2021**, *13*, 2234. [CrossRef]
108. Lemay, A.; Fernandez, J.M. Providing {SCADA} network data sets for intrusion detection research. In Proceedings of the 9th Workshop on Cyber Security Experimentation and Test ({CSET} 16), Austin, TX, USA, 8 August 2016.
109. Frazão, I.; Abreu, P.H.; Cruz, T.; Araújo, H.; Simões, P. Denial of service attacks: Detecting the frailties of machine learning algorithms in the classification process. In Proceedings of the International Conference on Critical Information Infrastructures Security, Kaunas, Lithuania, 24–26 September 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 230–235.
110. Pcap4J: A Java Library for Capturing, Crafting, and Sending Packets. Available online: Https://github.com/kaitoy/pcap4j (accessed on 31 May 2021).

111. Infrastructure, C. Threat Information Sharing Framework. A Reference Guide for the Critical Infrastructure Community. USA Homeland Security Available online: https://www.cisa.gov/sites/default/files/publications/ci-threat-information-sharing-framework-508.pdf (Accessed on 10 January 2022)
112. Saharkhizan, M.; Azmoodeh, A.; Dehghantanha, A.; Choo, K.K.R.; Parizi, R.M. An ensemble of deep recurrent neural networks for detecting iot cyber attacks using network traffic. *IEEE Internet Things J.* **2020**, *7*, 8852–8859. [CrossRef]
113. Anthi, E.; Williams, L.; Rhode, M.; Burnap, P.; Wedgbury, A. Adversarial attacks on machine learning cybersecurity defences in Industrial Control Systems. *J. Inf. Secur. Appl.* **2021**, *58*, 102717. [CrossRef]
114. Ayodeji, A.; Liu, Y.k.; Chao, N.; Yang, L.q. A new perspective towards the development of robust data-driven intrusion detection for industrial control systems. *Nucl. Eng. Technol.* **2020**, *52*, 2687–2698. [CrossRef]