energies

MDPI

Article

A Multi-Agent Adaptive Architecture for Smart-Grid-Intrusion Detection and Prevention

Tomasz Kisielewicz [1],*, Stanislaw Stanek [2] and Mariusz Zytniewski [3]

[1] Electrical Department, Warsaw University of Technology, 00-661 Warszawa, Poland
[2] Management Department, General Tadeusz Kosciuszko Military University of Land Forces, 51-147 Wroclaw, Poland; stanislaw.stanek@awl.edu.pl
[3] Department of Informatics, University of Economics in Katowice, 40-287 Katowice, Poland; mariusz.zytniewski@ue.katowice.pl
* Correspondence: t.kisielewicz@gmail.com

Abstract: The present paper deals with selected aspects of energy prosumers' security needs. The analysis reported aim to illustrate the concept of the implementation of intrusion-detection systems (IDS)/intrusion-prevention systems (IPS), as supporting agent systems for smart grids. The contribution proposes the architecture of an agent system aimed at collecting, processing, monitoring, and possibly reacting to changes in the smart grid. Furthermore, an algorithm is proposed to support the construction of a smart-grid-operating profile, based on a set of parameters describing the devices. Its application is presented in the example of data collected from the network, indicating the process of building a device-operation profile and a possible mechanism for detecting its changes. The proposed algorithm for building the operating profile of devices in the smart grid, based on the mechanism of continuous learning by the system, allows for detecting network malfunctions not only in terms of individual events but also regarding limits of the scope of system alerts, by determining the typical behavior of devices in the smart grid. The paper gives recommendations to a software-agent system development, which is dedicated to detecting and preventing anomalies in smart grids.

Keywords: safety and security; intrusion detection/prevention systems; software multi-agent systems

check for updates

Citation: Kisielewicz, T.; Stanek, S.; Zytniewski, M. A Multi-Agent Adaptive Architecture for Smart-Grid-Intrusion Detection and Prevention. *Energies* **2022**, *15*, 4726. https://doi.org/10.3390/en15134726

Academic Editors: Akhtar Kalam and Abu-Siada Ahmed

Received: 24 May 2022
Accepted: 24 June 2022
Published: 28 June 2022

1. Introduction

Security issues are an important strand of research in smart-grid theory, in particular for low-voltage smart-grid-system users. The entire system's security is largely contingent on how terminal units that effectively generate electricity are protected. A variety of software protections are employed to support the detection and prevention of illicit activity at different levels. The most common solution addressing this issue are intrusion-detection systems (IDS) [1,2]. An IDS is supposed to monitor the system, i.e., the grid and the users, and is typically geared to detecting any deviations from predefined rules and adopted indicator levels, where such deviations are interpreted as symptomatic of a potential attack on the system. The IDS framework is, therefore, often used in building support solutions for the management of smart-grid systems.

An IDS incorporated in a smart grid can monitor system behavior at a hardware level, where technical issues are detected, as well as at a software level, where the system's and the users' behavior patterns are examined. In the latter case, the IDS may be focused on the computer network handling data transfers, the software used by smart-grid devices, and user-behavior profiles. Essentially, it will be targeted at detecting events where systemic parameters deviate from an established norm. The subject literature describes a number of IDS types and implementations [3–5]. Two major IDS categories may be distinguished: Network Intrusion Detection Systems (NIDS) and Host Intrusion Detection Systems (HIDS). While a NIDS will analyze network traffic to identify anomalies in data packets being transmitted, an HIDS will observe incoming and outgoing traffic in terms of local users' activities and processes.

Furthermore, for security purposes, an intrusion-prevention system (IPS) shall be illustrated. An IPS can be treated as an extension of IDS, as it combines the use of mechanisms for detecting adverse behaviors with implementing specific security policies aimed at blocking activities that are interpreted as indicative of an attempted intrusion. In principle, IPS can perform a variety of actions, such as raising alarms, blocking suspect data packets, resetting connections, stopping all traffic, and so on [6].

The following IDS components can be characterized by [7]:

- Data-gathering device (sensor)—collecting data from systems being examined/protected;
- Detector (intrusion-detection engine)—the process that analyzes data from sensors for potential intrusions;
- Knowledge base (database)—definitions of sample attacks for use in identifying likely intrusions;
- Configuration device—mechanisms for determining current IDS settings and status;
- Response component—a mechanism for triggering actions to address likely threats detected.

Although this architecture is also referenced in several other studies [1,8], a closer look and a more in-depth analysis of its structure can be performed. In addition, a concept combining the three elements of IDS, namely a signature database, the detection of deviations from typical behaviors, and a user/device-profiling mechanism for optimum security of systems, can be applied [9].

The solution proposed in this paper enables to detect attacks like: physical intrusions, service spoofing, integrity violations, and bypassing controls, according to the typology proposed in [10].

As far as smart grids are concerned, the role of IDS/IPS is to ensure the security of the data processing taking place between the sensors, the data-update mechanism, and the control mechanism [11]. To enable IDS/IPS to cover such a wide application area and to adapt to emerging threats, it appears necessary to deploy artificial intelligence solutions. What these artificial intelligence algorithms should be able to do within IDS/IPS is not just collect and process data, but to also discover new knowledge on system behaviors and human behaviors.

Under this approach, a viable artificial intelligence solution should be supportive of the distributed nature of smart-grid systems, compliant with relevant information-exchange standards, capable of exercising autonomy in gathering and processing data, and easily scalable. That this article opts for and seeks to foster an agent-based approach to building IDS/IPS is predominantly attributable to the proactive characteristics of software agents, accounting for a system's ability to take initiative without having to rely on signals and prior data feeds from the environment. Another argument for the application of software-agent technologies stems from the distributed nature of multi-agent systems, which is most evident in the heterogeneous environments of organizational-information systems; hence, adding a new component to the existing architecture involves the use of just another agent unit, which makes such systems highly scalable. Multiple IPS/IDS architectures built around multi-agent systems are reported in [12–17].

The paper's goal is to illustrate the concept of IDS/IPS systems' implementation as supporting agent systems for smart grids, on the basis of a selected use-case description.

2. Reference-System Architecture

The changes in power systems indicate the need to include the generic structure of the response system, including a response component, detection component, monitoring component, and the management system, as well as an adaptation component based on the dynamic profiling of the elements essential for the functioning of the system. An extended generic view of an intrusion-response system is shown in Figure 1.

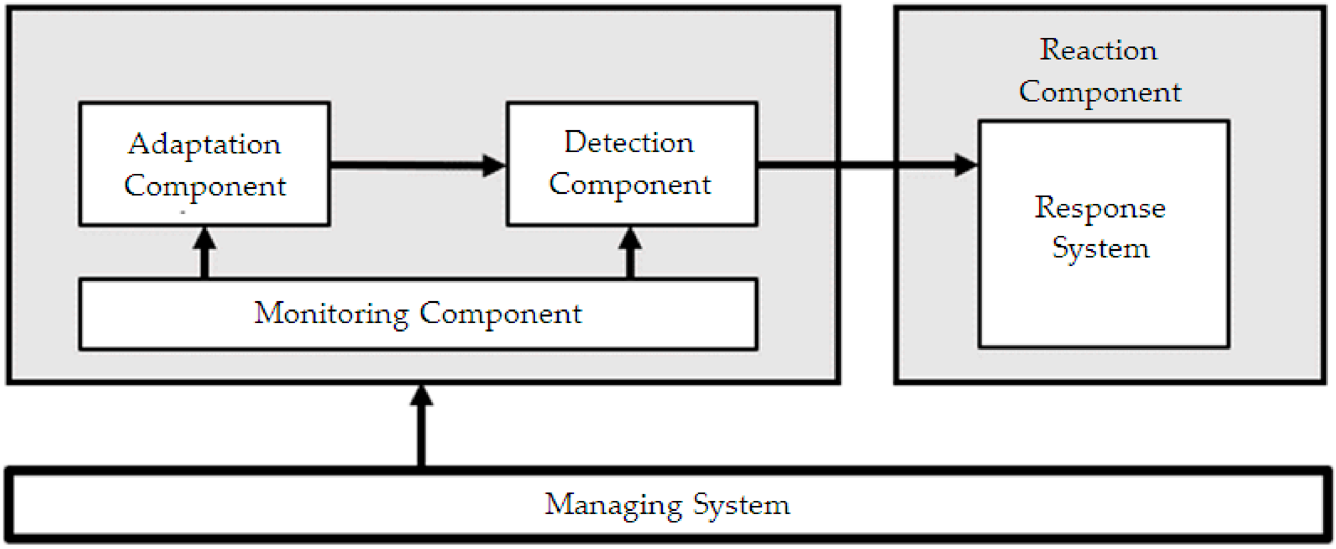


Figure 1. Intrusion response system: an extended generic view.

The literature review in the area of a response system shows that there are no solutions to the problem of intelligent adaptation, in particular in the context of power-grid systems and solutions based on multi-agent systems [18–25]. In addition to a number of other advantages, multi-agent technology supports the possibility of constructing friendly anthropomorphic user interfaces, which is important in the case under consideration.

The first step in designing IDS/IPS for use in a smart grid is to define its coverage. It is possible to distinguish home-area networks, neighborhood-area networks, and wide-area networks, where data relate to energy costs, production, and consumption and where devices such as a smart-meter-data collector (SMDC), central-access controller (CAC), supervisory-control and data-acquisition (SCADA) controller, energy-distributed system (EDS), etc., are found [10].

Within this kind of architecture, the most important factor is the low-level security of system-wide communication, which can be implemented via encryption, tokens, and passwords. However, on top of that, systems will call for control and profiling functions based on data analysis performed by artificial intelligence mechanisms, so that, in the event of security breach, it is possible to classify certain operations as potentially hazardous. It has already been observed that the use of agent-based technologies can offer multiple benefits in terms of the system's scalability and fault tolerance. Table 1 presents software agent applications in smart-grid development, taking into account different types of agents used, namely: monitoring agent, control agent, and supervisor agent [26]; management agent, relay agents, distributed-generator agents, and equipment agents [27]; communication-control agent, network-topology-analysis agent, impedance-matrix-computation agent, adaptive-setting-calculation agent, relay agent, mobile-management agent, and network-monitoring agent [28]; relay agent, configuration agent [29]; smart agent (circuit-breaker agents), and master agent [30]; and bus agents [31].

The application of software-agent theory to IDS/IPS development is linked, on the one hand, to the envisaged system architecture and, on the other, to the data assessment mechanisms used to identify potential threats. The most common approach is to use a signature database that contains definitions, e.g., of packets, indicative of a specific type of attack. This approach, while effective in detecting events that have already occurred in the past, may fail when a certain type of attack is launched for the first time. Therefore, another approach is that of predictive algorithms, associated with user-behavior profiling, to delineate a range of behaviors considered normal. This approach enables the system to adapt to changing environmental conditions. In case incidents are detected that show a statistically significant deviation from an adopted standard, the system takes predefined measures. User behaviors can be assessed and profiled, owing to the availability of a database of potential attack patterns and an algorithm for their investigation.

Table 1. Sample software-agent applications in smart-grid development.

Types of Agents Used	IDS/IPS System Architecture	Agent Platform Used	Simulation	Role of Agent System
Monitoring agent, control agent, supervisor agent	None	Information unavailable	Yes	Ground and line fault
Management agent, relay agents, distributed-generator agents and equipment agents	None	JADE	No	Coordination of communication in the system
Communication-control agent, network-topology analysis agent, impedance-matrix-computation agent, adaptive-setting-calculation agent, relay agent, mobile-management agent, network-monitoring agent	Dispatching Center, SDH-based Intranet, Station	Information unavailable	Yes	Adaptive-relay settings calculation
Relay agent, configuration agent	Multi-agent-communication layer, data-acquisition layer (PMU), physical-power-grid model	JADE	Yes	System reconfiguration, relay coordination
Smart agent (circuit-breaker agents), master agent	Agent, environment	JADE	Yes	Phase-fault detection
Bus agents	Multi-agent layer, physical layer	JADE	Yes	Fault diagnosis

While the relevance of artificial intelligence in IDS/IPS design has been demonstrated by a number of works [32–36], the limitations emphasized by [37] gives a proposal of a system/user assessment mechanism where the user-data-protection algorithm involves the use of software agents. In addition, the results reported in [37] showed that the method can be effective in building user-behavior profiles. The artificial intelligence algorithm proposed in this article is an extension of that approach and is to be applied in profiling the operations of a smart grid.

The IDS/IPS architectural concepts outlined above entail, as a next step, defining the functionalities of a modern IDS/IPS dedicated for smart-grid applications and supported by artificial intelligence mechanisms. In the first place, any such solution should take full account of the intrinsic business logic of information systems that are supported by translating their behavioral patterns into logical processes. In particular, such systems should be able to:

- Collect information on the operation of devices that are part of the system;
- Collect and process information about users and their activity in the system;
- Define and monitor the authorization levels of smart-grid users;
- Define and monitor the roles of smart-grid components and users;
- Define access rights to devices, their settings, and the data they generate;
- Monitor data flows in the computer network;
- Generate and monitor profiles defining the typical behaviors of devices and users;
- Apply system-performance risk-assessment mechanisms;
- Take measures to reduce this risk.

It should be borne in mind that the security of a smart grid does not correspond to the security of electricity flow alone or to coordinating the work of specific transmission elements. Instead, the security of such systems is to a large extent conditional on the security of the data processed by devices and the security of information about its users. Access to data on electricity consumption by residents can be used to pin down their daily routines, revealing when and which household members are at home.

Defining the unique logic of a system is an extremely challenging and time-consuming process. One way to tackle this task is to build the software from components, on a sort of modular frame, and to utilize software agents in doing so, with a distributed nature, proactivity, mobility, and communication capacity that make them an interesting alternative suited to the development of modern-day IDS/IPS. Another important aspect is a proactive-assessment mechanism for events that step beyond what is stored in a typical IDS/IPS knowledge base. Such databases are typically focused on network-traffic analysis and associated with the use of a specific traffic-analysis device. When developed in that way, the knowledge base is in fact attuned to a given device rather than to the heterogeneous corporate-information system. For this reason, the construction of advanced IDS/IPS involves the development of additional, dedicated knowledge bases on potential attacks, and the development of a solution capable of identifying potential intrusions based on behavior patterns specific to one or many information systems simultaneously. From this perspective, the primary issue here is how to capture all patterns of typical-user behavior, in what is referred to as a behavior-profile knowledge base.

The use of IDS/IPS systems in the field of smart grids requires the determination of a number of parameters related to the operation of such a system. The first one is related to the system architecture and the data flows between its modules. As will be shown later in the paper, the IDS/IPS systems' architectures, built in the area of the agent-based approach, do not show sufficient maturity to be used in a construction system allowing for the creation of a smart-grid-system operation profile, therefore, in Section 3, the proposed system architecture, which has previously been used in the construction of a system profiling the work of employees, will remain. Building a smart-grid operation profile (after collecting data on the operation of its devices) requires determining the algorithm that will be used to develop operating profiles of network devices. Section 4 presents a set of variables that were collected during the operation of the smart grid and shows and discusses the operation of the proposed algorithm. The results of the algorithm's operation are presented in Section 5, where the process of building two profiles of smart-grid operation is presented.

The analyzed case for data collection consists of set of automatic irrigation devices, namely a submersible pump, lighting, garden monitoring, and automation of the mowing process. The devices connected to the analyzed system include: a 1100 W submersible pump, a set of garden lamps with a total power of 300 W, a watering-mechanism control system, a video monitoring system for the garden, and a charging system for a mowing robot, with a total approximate power of 200 W. A schematic illustration of the analyzed system is shown in Figure 2.

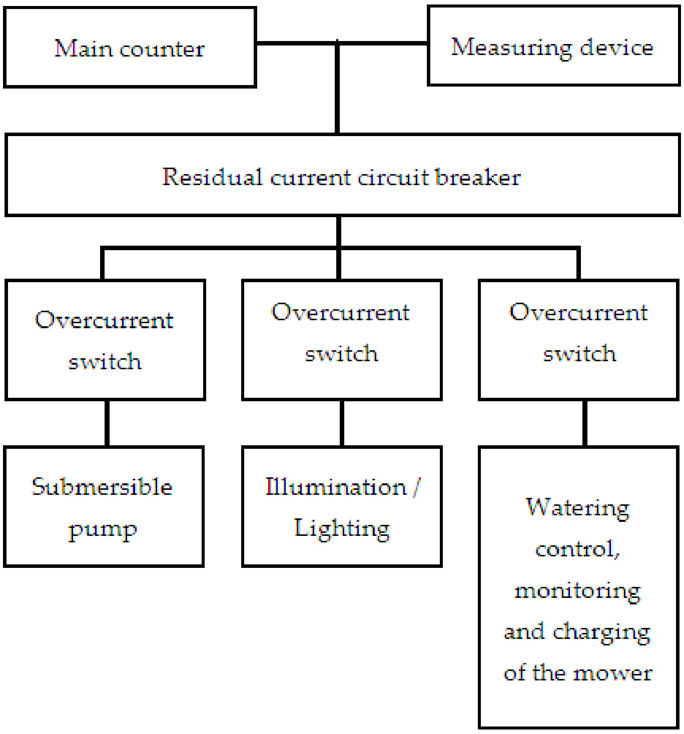


Figure 2. Schematic illustration of analyzed system for data collection.

3. An Approach to Deploy Software Agents in Risk Assessment and Protection of Information System Security

The present section gives an overview of the architecture of an IDS/IPS tailored to the requirements of organizations operating in heterogeneous environments. The proposed system architecture is shown in Figure 3.

The proposed system architecture is an extension of the reference IDS/IPS architecture found in [7]. First of all, there is no direct mention of a mechanism for system configuration and system status/self-check. This is due to the distributed nature of agent technologies. The functionality is provided by the multi-agent variety of JADE. Further, for an IDS/IPS that protects and examines multiple information systems, a single knowledge base may not suffice. Therefore, the proposed architecture has as many as four knowledge bases. The approach also differs in terms of tasks performed by agents: rather than merely collecting information about the operation of devices, in the proposed architecture agents will engage in data processing throughout the system.

The first module (Data gathering from devices) is responsible for accessing network devices and monitoring their condition. In the adopted architecture, each device is associated with a separate monitoring and data collecting agent. The multi-agent architecture allows for the use of multi-agent data exchange, specific for a given platform, thanks to which there is no need to define the communication protocols implemented in the system.

The second module (Detector) includes event profiling and monitoring, based on the data collected by the first module. The profiling process is related to the proposed algorithm presented in the next chapter and uses the data collected from the devices connected to such a system. The event-monitoring layer carries out the process of detecting changes from the prepared network-operation profiles. The events to be analyzed are subject to queuing and comparison with the developed system-operation profiles. The event warehouse is responsible for recording events related to the smart grid's day-to-day activities.

The third module (Response and risk assessment) is related to the concept of IPS systems and concerns building the rules of system behavior in the event of detecting received events from the prepared system-operation profiles.

The last module (Knowledge-warehouse layer) deals with the mechanisms, places of data storage, and rules processed by the system.

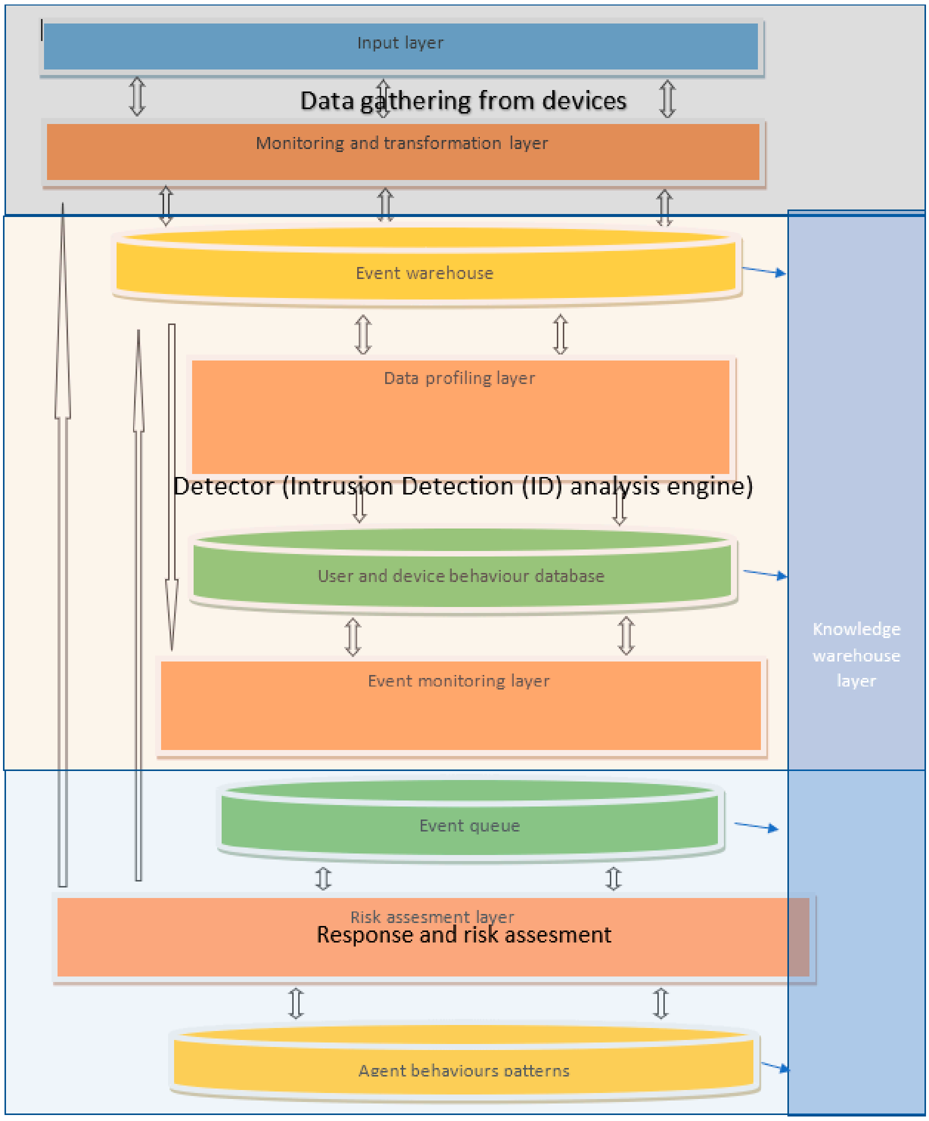

Input layer
Data gathering from devices
Monitoring and transformation layer
Event warehouse
Data profiling layer
Detector (Intrusion Detection (ID) analysis engine)
User and device behaviour database
Event monitoring layer
Knowledge warehouse layer
Event queue
Risk assesment layer
Response and risk assesment
Agent behaviours patterns

Figure 3. The reference structure of an agent-based IDS/IPS.

A distributed architecture founded on the use of software agents makes it possible to split individual system functions into subsystems, as shown in Figure 3. This alignment is further reflected in having separate knowledge bases for events, for user profiles, for event/message queuing, and for possible system responses to detected intrusions. It has already been pointed out that the choice of this modular architecture can be traced back to the diversity of systems that the data are gathered from, and to the system's commitment to exchanging data through messages, which represents a superb scalability and fault tolerance. A schematic illustration of the proposed system operation is shown in Figure 4.

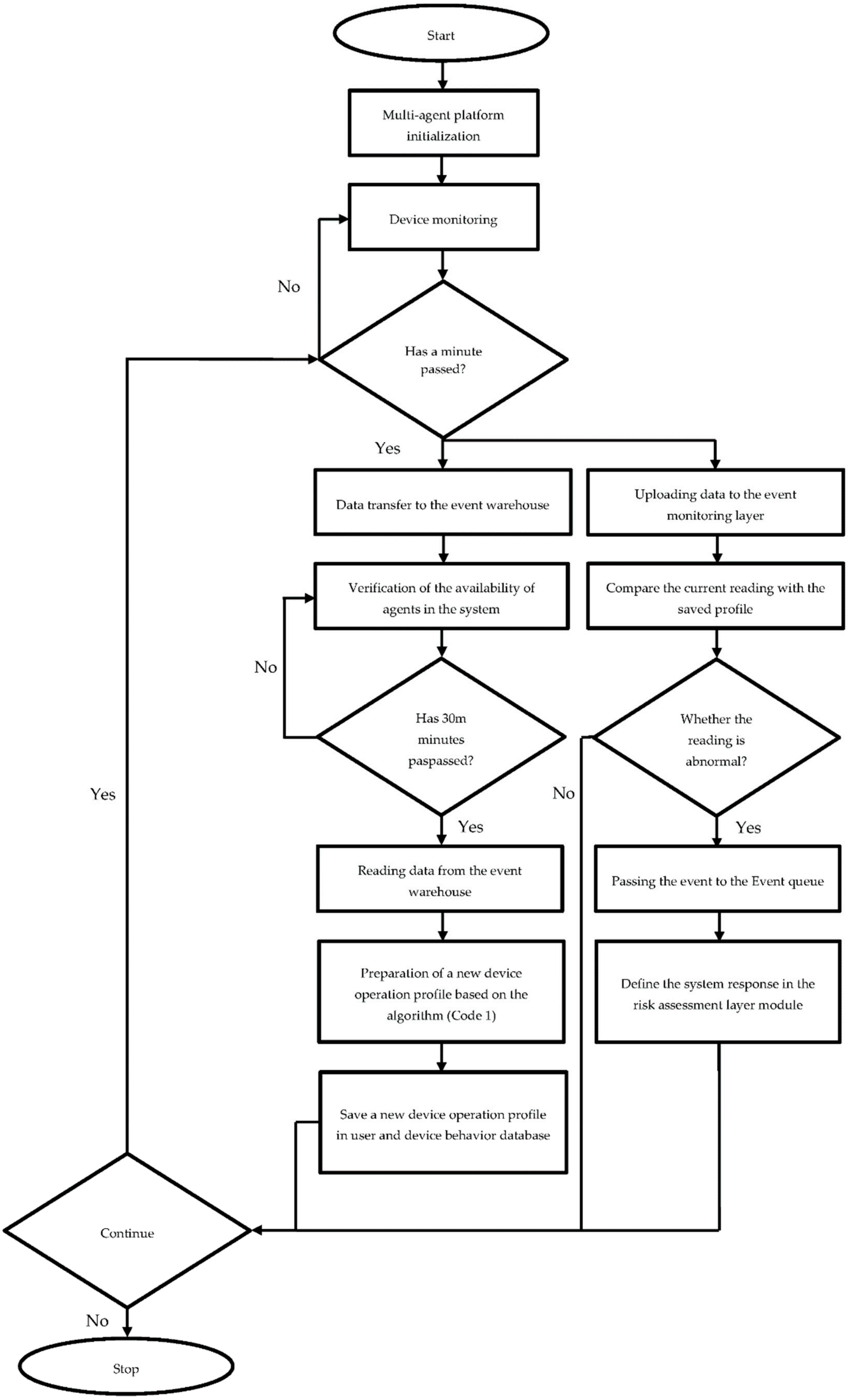

Start
Multi-agent platform initialization
Device monitoring
No
Has a minute passed?
Yes
Data transfer to the event warehouse
Uploading data to the event monitoring layer
Verification of the availability of agents in the system
Compare the current reading with the saved profile
No
Has 30m minutes paspassed?
Whether the reading is abnormal?
Yes
No
Yes
Yes
Reading data from the event warehouse
Passing the event to the Event queue
Preparation of a new device operation profile based on the algorithm (Code 1)
Define the system response in the risk assessment layer module
Save a new device operation profile in user and device behavior database
Continue
No
Stop

Figure 4. Multi-agent platform diagram.

4. Algorithms for User- and Device-Behavior Profiling

Section 2 brought attention to the problem faced by organizations in protecting personal data and detecting events that might lead to their loss. In the presented approach, the use of IDS/IPS cannot be based solely on network-traffic analysis, since the processing of personal data in information systems takes place on a permanent basis and involves computer networks. Hence, it is necessary to monitor whether any data are actually being transferred and, at the same time, build user (behavior) profiles to be able to capture activities that overstep the boundaries of what is acceptable. For such profiles to be built, the data on user behaviors must first be collected, a profiling algorithm must be developed, a profile storage mechanism must be set up, and a mechanism for analyzing and assessing the activities of users and smart-grid devices must be in place. The basic operating parameters of a smart grid include:

- **On-grid/off-grid**—information on system operation. The on-grid installation is disabled in the event of power failure. The off-grid installation uses its own batteries for electricity storage (True/False);
- **AC circuit break**—AC line protection mechanism (True/False);
- **DC circuit break**—DC line protection mechanism (True/False)
- **Energy generation**—amount of energy generated (kWh);
- **Energy consumption**—amount of energy consumed by the user (kWh);
- **AC mains voltage**—determines current voltage in the power grid; on-grid requires that the inverter cut off power transfer to the smart grid in the event that a certain value is exceeded (V);
- **DC mains voltage**—determines voltage in the photovoltaic system or another power-generation device (V);
- **Energy generated and transferred to grid**—amount of energy transferred to the power grid (kW);
- **Energy generated and consumed by user**—amount of energy consumed by the user (kW);
- **System status**—determines whether the inverter is enabled or disabled (on/off) (True/False).

Alongside these parameters, cost variables may be taken into account, which makes more sense with off-grid systems that are capable of storing energy within them. The model could include such variables as the price of energy consumed per kWh, the price of a kWh of energy produced, or the cost of energy storage per kW. These are not shown in our examples, as they do not immediately affect system security. However, they could be easily brought into the model should they be seen as relevant and applicable in, e.g., analyzing the economic safety of a smart-grid user. Figure 5 gives an overview on the smart-grid profile-building process.

The methodology is based on the following stages:

- **Stage 1**—data are gathered via devices that monitor electricity flows in the grid and upload the data to the server running the Open Source MQTT broker. The project uses the Eclipse Mosquitto server that integrates events implemented in the smart grid and allows for the construction of smart home systems. The data are then saved by an agent in the system database.
- **Stage 2**—data collected in stage 1 are processed and a set of descriptive statistics is generated. As a result, the key characteristics of the data can be determined and any irregularities detected, such as, for example, missing datasets (a fact that may be ascribed to device failure or power outage).
- **Stage 3**—based on the data processed in stage 2, a system profile is generated in line with the algorithm presented below (code 1). As new data streams continue to arrive, the profile is subject to ongoing adjustments and is updated on a regular basis.
- **Stage 4**—the profile created in previous iterations is used to monitor the grid and to detect any anomalies in the new, incoming data. Once the new data are verified

against the profile, the system recalculates the profile's parameters and updates them, and then the whole cycle is repeated.

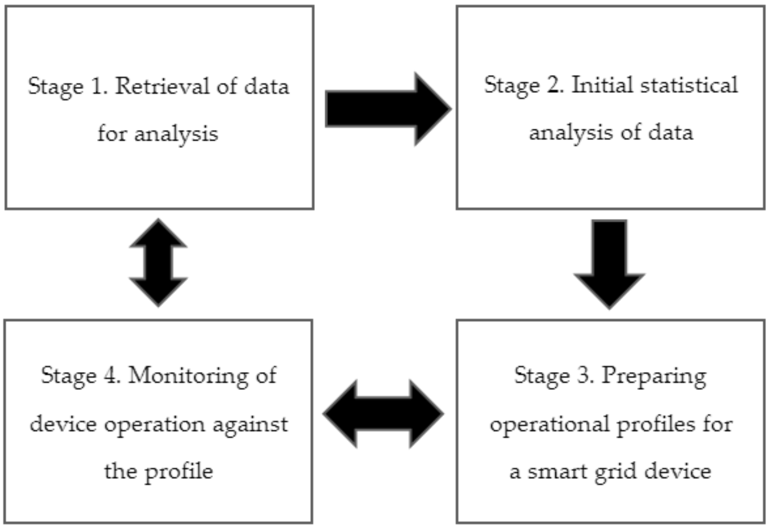

Stage 1. Retrieval of data for analysis
Stage 2. Initial statistical analysis of data
Stage 3. Preparing operational profiles for a smart grid device
Stage 4. Monitoring of device operation against the profile

Figure 5. Simplified illustration of the system operation.

Device profiling in a smart grid involves the development of an algorithm for a profiling agent (stage 3). The algorithm (Algorithm 1), if run repeatedly at regular intervals, should be able to build up a dataset describing the device's typical behaviors, thus allowing assessment of its operations under varying conditions.

Algorithm 1. Simplified algorithm for a profiling agent.

```
Code 1 ————————————————————————————————————————————
E = {$E_1$, ... ,$E_n$},//devices connected to the smart-grid network and devices supplied by the power system;
En.producer—identifier of the energy-producing device n;
En.client—identifier of the energy-consuming device n;
En.time—adopted unit of time;                                                  A
En.energy—amount of energy in kWh
P = {$P_1$, ... $P_m$}—set of behavior patterns
Pm = {$P_mE_n$.client,$P_mE_n$.energyMax, $P_mE_n$.energyMin, $P_mE_n$.timeFrom, $P_mE_n$.timeTo }
        //device-operation profiles over time;                                   B
AC.true,DC.true—correct operation confirmation;                                C
for each En∈E do
if En.time + 7 days < today do
for each Pm∈P do
PmEn.energyMax = null//maximum-energy consumption for the i device;
PmEn.energyMin = null//minimum-energy consumption for the i device;
      If En.client == PmEn.client && AC.true && DC.true && En.time >= PmEn.timeFrom
    && En.time <= PmEn.timeTo then
        If En.energy > PmEn.energyMax then
            PmEn.energyMax = En.energy;
        end if
        If Ei.energy <Ei.energyMin then
            PmEn.energyMin = En.energy;
        end if                                                                   D
Pm.add(En.client,PmEn.energyMax,PmEn.energyMin,PmEn.timeFrom,PmEn.timeTo);     E
    end if
end
end if
end
```

In the algorithm (A), it is proposed that each element of the smart grid should be described with the information regarding whether it is a producer or a consumer of electricity, in what time unit the current flow is measured, and how much electricity is consumed/supplied by a given device. In terms of the defined profile (B), the proposed algorithm indicates that it concerns the selected type of device, the maximum and minimum current supplied or consumed by this device, and the time range for which the consumption or current consumption profile is built. In the example shown below, the current consumption/supply profile was for 30 min of operation. Additionally, it was assumed that there is a current flow in the network (C). The section (D) shows the process of building a system-performance profile. For building a profile, it is necessary to define the time window in which the system will build the device profile. The algorithm assumes that it will be the period of the last 7 days. This window changes from day to day. If during the analyzed time there were no problems with powering the device (for a given profile of the analyzed device), the system checks whether the analyzed device's operation time applies to the time interval assigned to the profile being built. If so, then for a given time period in the profile, the system analyzes the minimum/maximum power consumption/production in this time period and assigns them to the profile. As a result, a set of profiles is built for a given device, indicating the maximum/minimum consumption/production of electricity in 30 min time windows during the last 7 days of system operation (E).

5. Application of the Proposed Architecture and Algorithm

The algorithm makes it possible to calculate any energy-consumption profile by a device connected to the smart grid. A profile can be built for, e.g., a whole building, as well as for a single device, e.g., a lamp. Figure 6 illustrates the operation of a JADE system that returns the outcomes shown.

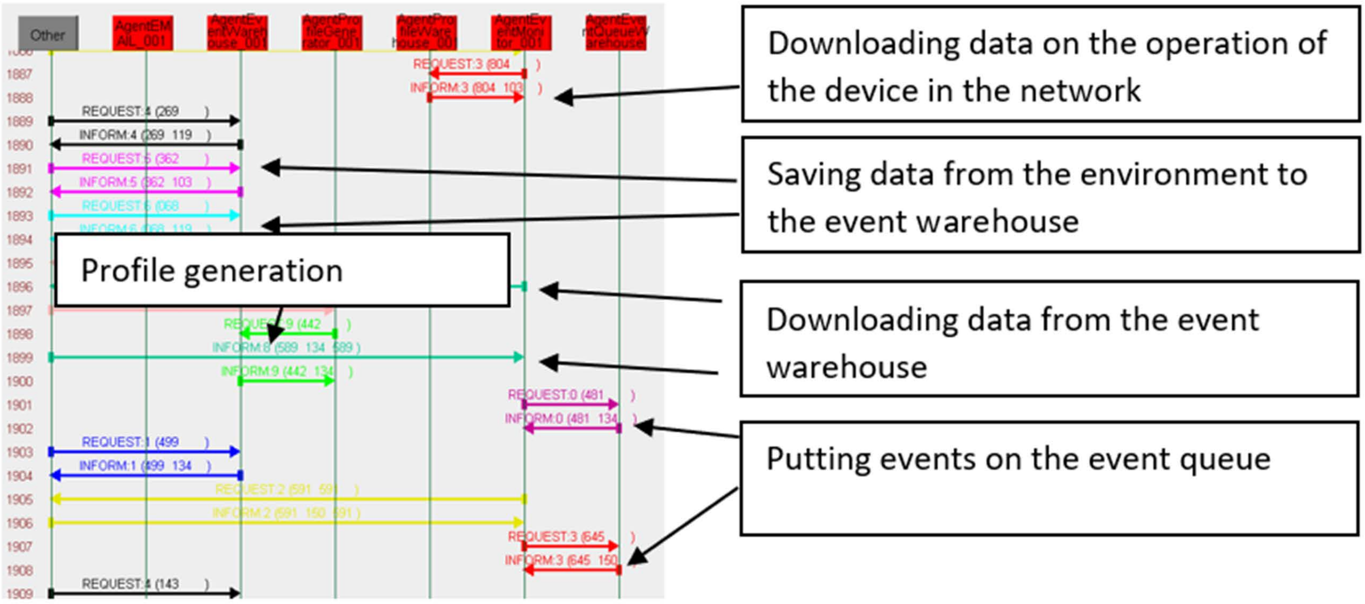
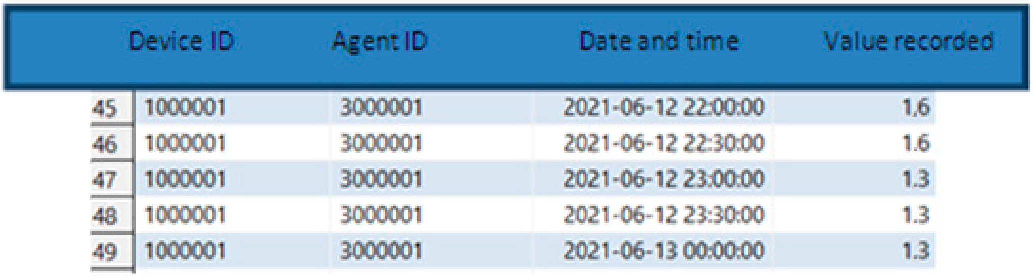


Figure 6. JADE platform at work.

In this example, data were collected on an hourly basis, which is typical for low-voltage smart-grid devices. Each time the profile-building process is relaunched, the system analyzes another day's data and produces a new profile. As a result, it is possible to analyze the profile's variability across subsequent days and hours. The user profile employs such statistical variables as mean value and standard deviation.

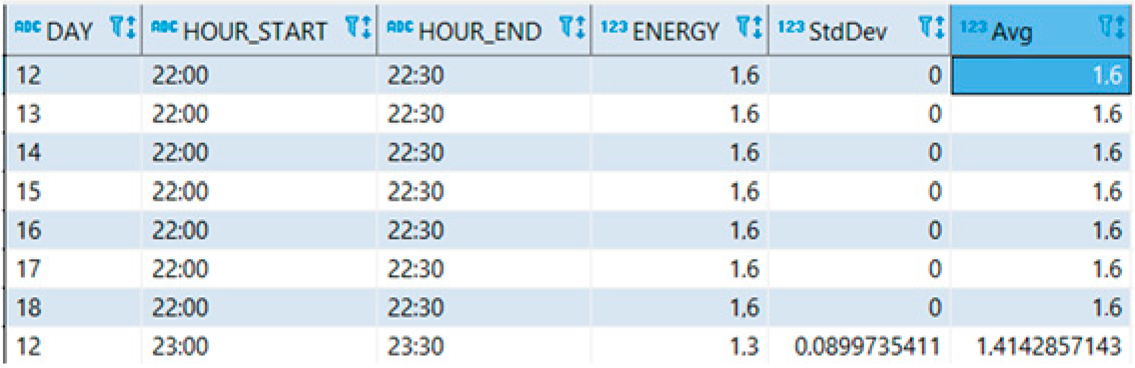
Figure 7 shows an example of raw data after they have been saved in the read database. In this example, the data were recorded at 30 min intervals. Figure 4 shows the resulting descriptive statistics.



	Device ID	Agent ID	Date and time	Value recorded
45	1000001	3000001	2021-06-12 22:00:00	1.6
46	1000001	3000001	2021-06-12 22:30:00	1.6
47	1000001	3000001	2021-06-12 23:00:00	1.3
48	1000001	3000001	2021-06-12 23:30:00	1.3
49	1000001	3000001	2021-06-13 00:00:00	1.3

Figure 7. Stage 1. Raw data for electricity consumption.

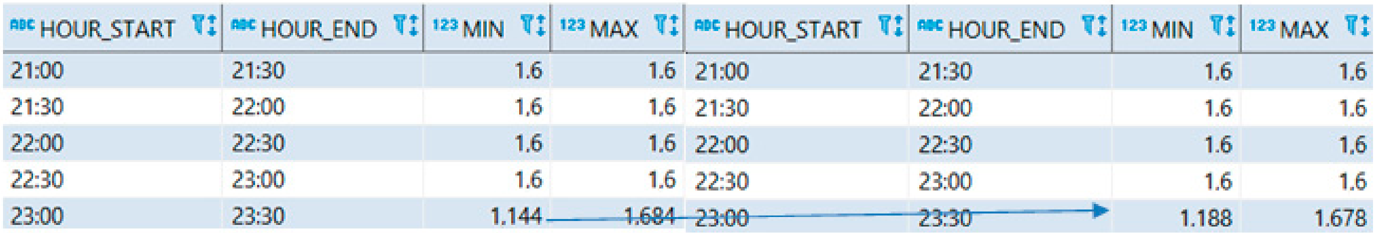
It is easy to see that (Figure 8) the typical energy consumption for the garden appliances averaged 1.6 kWh between 23:00 p.m. and 23:30 p.m. over the period examined. The algorithm helped detect an anomaly, indicating a drop in electricity consumption to 1.3 kWh on day 12 between 11 p.m. and 11:30 p.m. The standard deviation in this case was 0.09 kWh. The difference may have been due to an intentional shutdown (planned savings) just as well as a breakdown. If the theory of normal distribution is applied to the 2 days of data, with an accuracy of 99.7%, it should yield variance in the electricity consumption between 0.144 kWh and 1.684 kWh. Therefore, the decrease has to be regarded as incidental and should not trigger an alarm.



DAY	HOUR_START	HOUR_END	ENERGY	StdDev	Avg
12	22:00	22:30	1.6	0	1.6
13	22:00	22:30	1.6	0	1.6
14	22:00	22:30	1.6	0	1.6
15	22:00	22:30	1.6	0	1.6
16	22:00	22:30	1.6	0	1.6
17	22:00	22:30	1.6	0	1.6
18	22:00	22:30	1.6	0	1.6
12	23:00	23:30	1.3	0.0899735411	1.4142857143

Figure 8. Stage 2. Statistics for raw data on consecutive days.

As shown in Figure 9, the data-analysis window is altered on each subsequent day covered by the study. The changes are minor but demonstrative of the system's ability to adjust to changes in user activities. At the same time, the system can autonomously modify the profile and record changes in the behavior of devices.



HOUR_START	HOUR_END	MIN	MAX	HOUR_START	HOUR_END	MIN	MAX
21:00	21:30	1.6	1.6	21:00	21:30	1.6	1.6
21:30	22:00	1.6	1.6	21:30	22:00	1.6	1.6
22:00	22:30	1.6	1.6	22:00	22:30	1.6	1.6
22:30	23:00	1.6	1.6	22:30	23:00	1.6	1.6
23:00	23:30	1.144	1.684	23:00	23:30	1.188	1.678

Figure 9. Changes in user-behavior profile on consecutive days.

6. Conclusions

The results give an overview of a multi-agent adaptive architecture for smart-grid-intrusion detection and prevention, in light of service continuity and prosumers' protection needs. The proposed approach for multi-agent adaptive-architecture development can diminish the likelihood of personal-data leakage, through monitoring and profiling user activities.

The paper's quantitative results demonstrates that the use of software-agent technology has an influence on:

- process automatization of user-activity profiling and supervision;
- reduction possibility of data leakage;
- scalability and efficiency improvement of the system, through the use of multiple instances of agents simultaneously performing associated tasks;
- improvements in mitigating the risks of data leakage or account-takeover fraud (i.e., taking over an employee's login credentials as a result of phishing attacks and spamming).

In addition, the profiling mechanism, based on statistical analysis, can easily adapt to changing user preferences.

Author Contributions: Conceptualization, T.K. and S.S.; Formal analysis, S.S. and M.Z.; Investigation, S.S. and M.Z.; Methodology, S.S. and M.Z.; Supervision, T.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by the Warsaw University of Technology, General Tadeusz Kosciuszko Military University of Land Forces, and University of Economics in Katowice.

Acknowledgments: The paper has been prepared in the frame of interdisciplinary cooperation. The authors wish to express their gratefulness to the authorities of the Warsaw University of Technology, General Tadeusz Kosciuszko Military University of Land Forces, and University of Economics in Katowice for their relative support.

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

References

1. Denning, D.E. An Intrusion-Detection Model. *IEEE Trans. Softw. Eng.* **1987**, *SE-13*, 222–232. [CrossRef]
2. Anderson, J.P. *Computer Security Threat Monitoring and Surveillance*; Technical Report; James P. Anderson Co.: Ambler, PA, USA, 1980.
3. Debar, H.; Dacier, M.; Wespi, A. Towards a taxonomy of intrusion-detection systems. *Comput. Netw.* **1999**, *31*, 805–822. [CrossRef]
4. Wu, S.X.; Banzhaf, W. The use of computational intelligence in intrusion detection systems: A review. *Appl. Soft Comput.* **2010**, *10*, 1–35. [CrossRef]
5. Vasilomanolakis, E.; Karuppayah, S.; Mühlhäuser, M.; Fischer, M. Taxonomy and Survey of Collaborative Intrusion Detection. *ACM Comput. Surv.* **2015**, *47*, 1–33. [CrossRef]
6. Boyles, T. *CCNA Security Study Guide: Exam 640-553*; John Wiley and Sons: Hoboken, NJ, USA, 2010.
7. Sabahi, F.; Movaghar, A. Intrusion Detection: A Survey. In Proceedings of the 2008 Third International Conference on Systems and Networks Communications, Sliema, Malta, 26–31 October 2008; pp. 23–26. [CrossRef]
8. Pez, R.; Páez, R. An Agent Based Intrusion Detection System with Internal Security. In *Intrusion Detection Systems*; InTech: Houston, TX, USA, 2011. [CrossRef]
9. Peng, J.; Choo, K.-K.R.; Ashman, H. User profiling in intrusion detection: A review. *J. Netw. Comput. Appl.* **2016**, *72*, 14–27. [CrossRef]
10. Ullah, I.; Mahmoud, Q.H. An intrusion detection framework for the smart grid. In Proceedings of the Canadian Conference on Electrical and Computer Engineering, Windsor, ON, Canada, 30 April–3 May 2017; pp. 1–5. [CrossRef]
11. Gamage, T.T.; Roth, T.P.; McMillin, B.M. Confidentiality Preserving Security Properties for Cyber-Physical Systems. In Proceedings of the International Computer Software and Applications Conference, Munich, Germany, 18–22 July 2011; pp. 28–37. [CrossRef]
12. Albers, P.; Camp, O.; Percher, J.-M.; Jouga, B.; Mé, L.; Puttini, R.S. Security in Ad Hoc Networks: A General Intrusion Detection Architecture Enhancing Trust Based Approaches. In Proceedings of the Wireless Information Systems (WIS 2002), Ciudad Real, Spain, 2–3 April 2002; pp. 1–12.
13. Krmicek, V.; Celeda, P.; Rehak, M.; Pechoucek, M. Agent-Based Network Intrusion Detection System. In Proceedings of the 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, Fremont, CA, USA, 2–5 November 2007.
14. Ganapathy, S.; Yogesh, P.; Kannan, A. Intelligent Agent-Based Intrusion Detection System Using Enhanced Multiclass SVM. *Comput. Intell. Neurosci.* **2012**, *2012*, 9. [CrossRef] [PubMed]
15. Abdurrazaq, M.N.; Bambang, R.T.; Rahardjo, B. Distributed intrusion detection system using cooperative agent based on ant colony clustering. In Proceedings of the 2014 International Conference on Electrical Engineering and Computer Science (ICEECS), Kuta, Bali, Indonesia, 24–25 November 2014; pp. 109–114. [CrossRef]
16. Banik, S.M.; Pena, L. Deploying agents in the network to detect intrusions. In Proceedings of the 2015 IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS), Las Vegas, NV, USA, 28 June–1 July 2015; pp. 83–87. [CrossRef]

17. Zytniewski, M. Gossip and Ostracism in Modelling Automorphosis of Multi-agent Systems. In *Complexity in Information Systems Development*; Goluchowski, J., Pankowska, M., Linger, H., Barry, C., Lang, M., Schneider, C., Eds.; Lecture Notes in Information Systems and Organisation; Springer: Cham, Switzerland, 2017; Volume 22, pp. 135–150. [CrossRef]
18. Stakhanova, N.; Basu, S.; Wong, J. A taxonomy of intrusion response systems. *Int. J. Inf. Comput. Secur.* **2007**, *1*, 169–184. [CrossRef]
19. Available online: https://www.sciencedirect.com/topics/computer-science/intrusion-response-system (accessed on 24 May 2022).
20. Anwar, S.; Zain, J.M.; Zolkipli, M.F.; Inayat, Z.; Khan, S.; Anthony, B.; Chang, V. From Intrusion Detection to an Intrusion Response System: Fundamentals, Requirements, and Future Directions. *Algorithms* **2017**, *10*, 39. [CrossRef]
21. Inayat, Z.; Gani, A.; Anuar, N.B.; Khan, M.K.; Anwar, S. Intrusion response systems: Foundations, design, and challenges. *J. Netw. Comput. Appl.* **2016**, *62*, 53–74. [CrossRef]
22. Rullo, A.; Serra, E.; Lobo, J. Redundancy as a Measure of Fault-Tolerance for the Internet of Things: A Review. In *Policy-Based Autonomic Data Governance*; Calo, S., Bertino, E., Verma, D., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2019; Volume 11550. [CrossRef]
23. Aldaej, A. Enhancing Cyber Security in Modern Internet of things (IoT) Using Intrusion Prevention Algorithm for IoT (IPAI). In *IEEE Access*; IEEE: Piscataway, NJ, USA, 2019. [CrossRef]
24. James, F. IoT Cybersecurity based Smart Home Intrusion Prevention System. In Proceedings of the 2019 3rd Cyber Security in Networking Conference (CSNet), Quito, Ecuador, 23–25 October 2019. [CrossRef]
25. Rullo, A.; Bertino, E.; Sacca, D. PAST: Protocol-Adaptable Security Tool for Heterogeneous IoT Ecosystems. In Proceedings of the 2018 IEEE Conference on Dependable and Secure Computing (DSC), Kaohsiung, Taiwan, 10–13 December 2018; pp. 1–8. [CrossRef]
26. Kato, T.; Kanamori, H.; Suzuoki, Y.; Funabashi, T. Multi-Agent based Control and Protection of Power Distributed System-Protection Scheme with Simplified Information Utilization. In Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems, Arlington, VA, USA, 6–10 November 2005. [CrossRef]
27. Wan, H.; Wong, K.; Chung, C. Multi-agent application in protection coordination of power system with distributed generations. In Proceedings of the 2008 IEEE Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century, Pittsburgh, PA, USA, 20–24 July 2008; pp. 1–6. [CrossRef]
28. Zhu, Y.; Song, S.; Wang, D. Multiagents-based wide area protection with best-effort adaptive strategy. *Int. J. Electr. Power Energy Syst.* **2008**, *31*, 94–99. [CrossRef]
29. Rahman, M.; Isherwood, N.; Oo, A. Multi-agent based coordinated protection systems for distribution feeder fault diagnosis and reconfiguration. *Int. J. Electr. Power Energy Syst.* **2018**, *97*, 106–119. [CrossRef]
30. Satuyeva, B.; Sultankulov, B.; Nunna, H.S.V.S.K.; Kalakova, A.; Doolla, S. Q-Learning based Protection Scheme for Microgrid using Multi-Agent System. In Proceedings of the 2019 International Conference on Smart Energy Systems and Technologies (SEST), Porto, Portugal, 9–11 September 2019; pp. 1–6. [CrossRef]
31. Tian, F.; Wen, F.; Wang, X.; Xue, Y.; Salam, A. A multi-agent system based fault diagnosis for active distribution systems. In Proceedings of the 2016 IEEE Innovative Smart Grid Technologies-Asia (ISGT-Asia), Melbourne, VIC, Australia, 28 November–1 December 2016; pp. 1110–1114. [CrossRef]
32. Ye, N.; Chen, Q. An anomaly detection technique based on a chi-square statistic for detecting intrusions into information systems. *Qual. Reliab. Eng. Int.* **2001**, *17*, 105–112. [CrossRef]
33. Ling, L.; Song, S.; Manikopoulos, C. Windows NT User Profiling for Masquerader Detection. In Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control, Ft. Lauderdale, FL, USA, 23–25 April 2006. [CrossRef]
34. Revett, K. A bioinformatics based approach to user authentication via keystroke dynamics. *Int. J. Control Autom. Syst.* **2009**, *7*, 7–15. [CrossRef]
35. Pannell, G.; Ashman, H. Anomaly detection over user profiles for intrusion detection. In Proceedings of the 8th Australian Information Security Management Conference, Perth, Australia, 30 November 2010.
36. Gupta, S.; Kumar, P.; Abraham, A. A Profile Based Network Intrusion Detection and Prevention System for Securing Cloud Environment. *Int. J. Distrib. Sens. Netw.* **2013**, *9*, 364575. [CrossRef]
37. Zytniewski, M.; Stanek, S. Software agents supporting the security of IT systems handling personal information. *J. Decis. Syst.* **2020**, *29*, 285–300. [CrossRef]