

Article

Intelligent Access Control Design for Security Context Awareness in Smart Grid

Hyounghu Kim ¹  and Junho Choi ^{2,*}

¹ Department of Computer Engineering, Chosun University, Gwangju 61452, Korea; snowlisakim@gmail.com

² Division of Undeclared Majors, Chosun University, Gwangju 61452, Korea

* Correspondence: xdman@chosun.ac.kr; Tel.: +82-62-230-7624

Abstract: Recently, damages such as internal system intrusion, network and device vulnerability attacks, malicious code infection, and information leakage due to security attacks are increasing within the smart grid environment. Detailed and dynamic access control must be implemented to enable the power system in the smart grid environment to respond to such attacks. Dynamic and partial delegation must be available, and permission role restrictions must be considered for dynamic access control when delegating a role because of changes in power resource manager authority. In this paper, we propose an intelligent access control framework that can recognize security context by analyzing security vulnerabilities for security management of power systems. The intelligent access control framework is designed as a framework that enables collaboration within the smart grid environment, and a system administrator is designed to transmit access control policy information required between the power service principal and the agent. In addition, an experiment is conducted for the control inference of security context ontology-based access, attack detection inference of the security context awareness service, and the attack response of the intelligent integrated access control system. Experimental results show that the precision of security context ontology-based access control inference is 70%, and the attack response rate of integrated access control is 72.8%.



Citation: Kim, H.; Choi, J. Intelligent Access Control Design for Security Context Awareness in Smart Grid. *Sustainability* **2021**, *13*, 4124. <https://doi.org/10.3390/su13084124>

Academic Editors: O-Joun Lee and Hoon Ko

Received: 8 February 2021

Accepted: 5 April 2021

Published: 7 April 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: smart grid security; ontology inference; intelligent access control

1. Introduction

Smart grids are applied with various new IT technologies such as advanced metering infrastructure (AMI), IoT technology, smart meter technology, home area network (HAN), and cloud computing. However, the smart grid is exposed to various cyberattacks. And when it receives a security attack, it causes massive national damage [1]. Since attack methods for smart grids are becoming more diverse and intelligent, integrated countermeasures are required. Attacks on the power system in the smart grid environment can be primarily classified into structural, physical, and external attacks. Structural attacks are attacks that use vulnerabilities in the architectural design of a system, such as attacks using protocols, authentication procedures, and weaknesses in system modularization [2]. Physical attacks are attacks that threaten physical assets such as power transformers, circuit breakers, smart meters, and cables. In addition, external attacks include attacks using Trojan horses, viruses, and worms. The power system requires detailed and dynamic access control to respond to these attacks. Dynamic and partial delegation must be available, and permission role restrictions must be considered for dynamic access control when delegating a role because of changes in power resource manager authority [3]. In addition, the purpose, conditions, and obligations for data access must be considered for the protection of power data, and access must be denied when necessary. Therefore, the dynamic access control framework has the essential functions of collecting and analyzing security situation information, and of managing security access control policies. Herein, an intelligent access control framework is proposed to understand the security context by analyzing the security vulnerability of the power system for the security management of the power system [4–6]. The intelligent

access control framework was designed as a framework that enables collaboration within the smart grid environment, and a system administrator was designed to transmit access control policy information required between the power service principal and the agent [7,8]. In addition, we analyzed the security vulnerability of the power system in the smart grid and built a security context ontology [9]. An optimized access control policy was created and evaluated by inferring the security context through the defined reasoning rules. Herein, investigations regarding the access control policy are summarized in Section 2, and an intelligent access control framework to understand the security context is proposed; its security context ontology and inference rules are defined in Section 3. The precision of system access authorization for the proposed intelligent access control framework and the attack response rate of integrated access control were evaluated experimentally, and the details are provided in Section 4.

2. Related Studies

Physical damage due to the effect of cyberattacks may occur on the cyber physical system because the power system in the smart grid is accompanied by physical control as well as communication with the supervisory control and data acquisition system and field devices. Security threats discovered in power devices in the smart grid include memory dump, attack through access to a communication port for management, unauthorized access, firmware vulnerability attack, attack by exploiting the vulnerability of remote firmware update function, impersonation, denial-of-service attack, and malicious code injection attack [10–12]. When comparing the power system in the smart grid and the conventional information system from a security perspective, it can be inferred that the application of security technology suitable for a closed network environment that considers the characteristics of the power system is essential because the power system is operated for a specific purpose. In the power system, the response speed and processing within the critical time of a request from a top server, or the event transmission and server request from a subordinate server are important factors. In other words, the integrity and availability of accurately transmitting information processed within a set time range are the most important elements of the power system security policy, and immediate processing is important for accepting errors or faults and ensuring the safety of control system operation [13,14]. Therefore, various security issues such as virtualization technology security, large-scale distributed processing technology, service availability, huge traffic handling, application security, access control, authentication, and encryption are relevant in the smart grid environment. In particular, an intelligent access control model that can perform integrated management and control is required when accessing various resources of the power system. Restricting access to authorized users using role-based access control (RBAC) or context-aware role-based access control (C-RBAC) is the representative access control method. However, dynamic access control is unavailable in RBAC as it does not consider the security context element, and C-RBAC cannot guarantee the confidentiality and integrity of information as it does not consider the security level between the objects to be accessed [15,16]. In addition, RBAC and C-RBAC cannot prevent information leakage through legitimate access by objects associated with the duty they are performing. Conventional access control models fail to provide an efficient and feasible solution to security problems, such as the violation of the principle of least privilege, separation of duties, and information leakage, which can occur during multilevel delegation. Therefore, a dynamic and useful intelligent access control model is required that can compensate for the weaknesses of the conventional models [17].

3. Design of Ontology-Based Intelligent Security Context Awareness Service

3.1. Overall Framework

The intelligent integrated access control system of the power system was designed as a framework that enables collaboration within the smart grid environment, and the system administrator was designed to deliver access control policy information required

between the power service principal and the agent. The proposed system consists of a security context information collection and analysis engine that provides security context information collection and analysis functions, a security context inference module for security context ontology and inference, and an integrated access control module that provides security policy configuration and management functions. The overall framework of the security context awareness service is as follows (Figure 1).

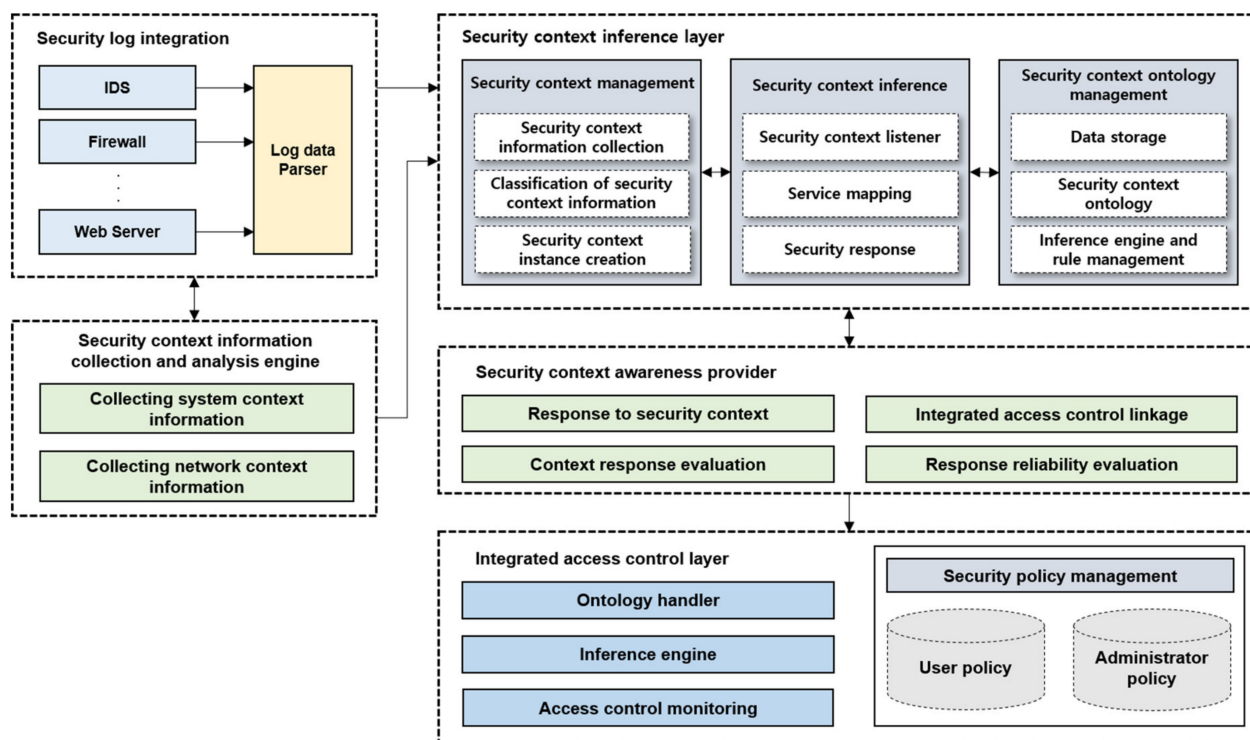


Figure 1. The overall framework of the security context awareness service.

When a security breach occurs in the power system, the agent is notified of the accident location, and information is updated. The analyzed attack pattern and attack intention are determined and reflected in the access control policy within the power system. The security context awareness service is arranged in the master server to monitor the security context of the power system, operates periodically in the host, and keeps the system resource efficiency constant for system performance. It maintains a certain system resource efficiency to prevent system performance degradation and generates security warnings when signs of security breach incidents are indicated through the security context awareness service. Subsequently, it performs security breach analysis. The security context inference layer collects, classifies, and creates security context instance information. It infers the security context using the transmitted security context information. The security context inference module uses the information defined in the ontology to determine the security context. The security context ontology management module generates a security context event in the security context inference module, after storing the security context information collected by the security context management module as a module that collects security context information. The inference engine and rule management module generate a query to infer the security context ontology and convert the security context ontology into web ontology language (OWL). The security context inference module infers the security context, after converting the transmitted security context information into a query that can be inferred by ontology. It refers to security context ontology and designed inference rules for inference. A response method for the security context is provided by providing the security context to the security context awareness provider layer after inferring the security context information. The security context instance generation module converts the low-

level security context information generated by the security context information collection and classification module into an instance of the security context ontology. Context-aware inference is performed by the security context inference module after the generated instance is stored in the ontology storage. The security context instance generation module applies the domain and upper ontology model suitable for the context through the security context service mapping module and generates the security context instance applied through the security context instance generation module. In regard to the security context ontology management module, the security context ontology model schema information, instances, and security context rules are stored in the data storage.

3.2. Operation Scenario of Security Context Awareness Service Framework

Figure 2 shows an operation scenario between components of the power system security context awareness service framework. The security context information collected from the internal system and the security system in the power system is delivered to the security context management module of the security context inference layer. Security context management converts the collected security context information into a form that can be used in the security context ontology management module.

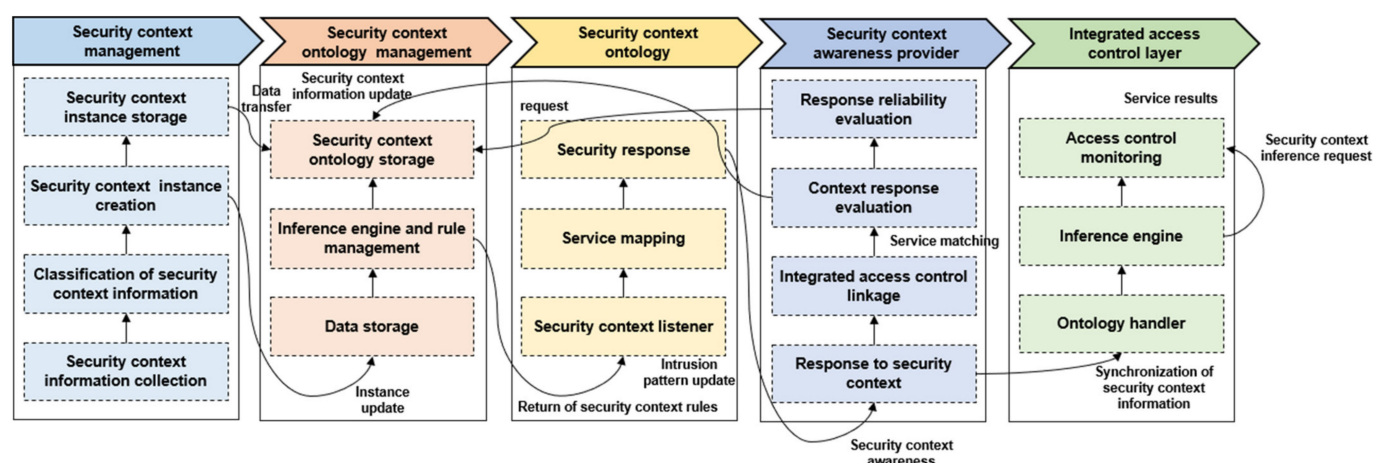


Figure 2. An operation scenario of power system security context awareness service.

The host and network context information of the power system is a factor that determines whether the network and system are running in a normal operating state. The system inside the power system operates abnormally when the system wait time suddenly increases due to a security attack or when the number of failed secure shell (SSH) login attempts is high. The security context information can confirm whether the response is effective when the system or network is actually being attacked. In the security context awareness service, the security situation analysis engine captures and analyzes security situation information in real time whenever a new intrusion is detected and a new intrusion warning is received. The network security context information is a snapshot of the traffic of the sub-network in the power system. It consists of the current time and date, protocol type, IP address, port, IP destination address, destination port, traffic parameters, received and transmitted bytes, and so on. The security context analysis engine analyzes the security context information in real time whenever it receives a new intrusion warning. System security context information consists of the number of active processes, CPU usage, free disk space, waiting time, number of recorded users, system status, number of SSH failed logins, and number of zombie processes.

The security context information transmitted from the security context information collection and classification module generates a new security context instance through the security context ontology management module. The converted security context instance is transmitted to the security context inference module for context inference and simul-

taneously updated by the internal security context ontology management module. The security context in the power system is inferred based on the security context instance and pattern transmitted from the security context instance generation layer of the security context inference module, and an appropriate service for the context is determined. The security context awareness provider delivers the inference result delivered through the security context inference layer to the integrated access control layer. And, the integrated access control layer requests a response according to the security policy.

3.3. Design of Security Context Ontology and Inference Rule

Attack items in the power system include internal system intrusion, network and device vulnerability, malicious code infection, and information leakage, which can be primarily classified into structural, physical, and external attacks. Structural attacks are attacks that use vulnerabilities in the architectural design of the system, such as attacks using weaknesses in protocols, authentication procedures, and system modularization. Physical attacks are attacks that use vulnerabilities in the source code, such as attacks using structured query language (SQL) injection and buffer overflow. External attacks are attacks that use programs other than the target of attack, such as attacks using Trojan horses, viruses, and worms [18,19]. These attacks are analyzed and used to build an ontology for security context information in the system. The security attack elements and details of the power system attacks are as shown in Table 1.

Table 1. Security attack elements and details of power system attacks.

Security Attack Element	Details of Security
Audit record	Inquire history of setting changes and security function executions, and log generation history; notify administrator in advance when capacity is exceeded, protect log and restrict access to log when the capacity is exceeded, and control log deletions and changes
Bot detection and response	Control various communication protocols, signature and behavior-based analysis, cure malicious bot-infected hosts, remove malicious bots, detect and block abnormal traffic, and notify abnormal traffic to administrator
Identification and authentication	Administrator identification and authentication, set a stronger password policy, masking when entering/changing password, and control error information in case of password failure
Transmitted data protection	Encrypt transmission when transmitting and receiving data; verify safety of encryption related protocols
Agent protection	Ensure execution file integrity, filter driver integrity, and recovery function when tampered information is discovered; guarantee non-repudiation and integrity

The system security context awareness service is provided through the inference process of the rule-based inference engine of ontology after acquiring security context information through the log structure analysis of various security systems in the power system. The use of ontology enables high-level information to be extracted from low-level information, generalizes the information, and defines the relationship between the information. The elements of the security context ontology are represented as classes in the hierarchical structure of OWL, as shown in Figures 3 and 4.

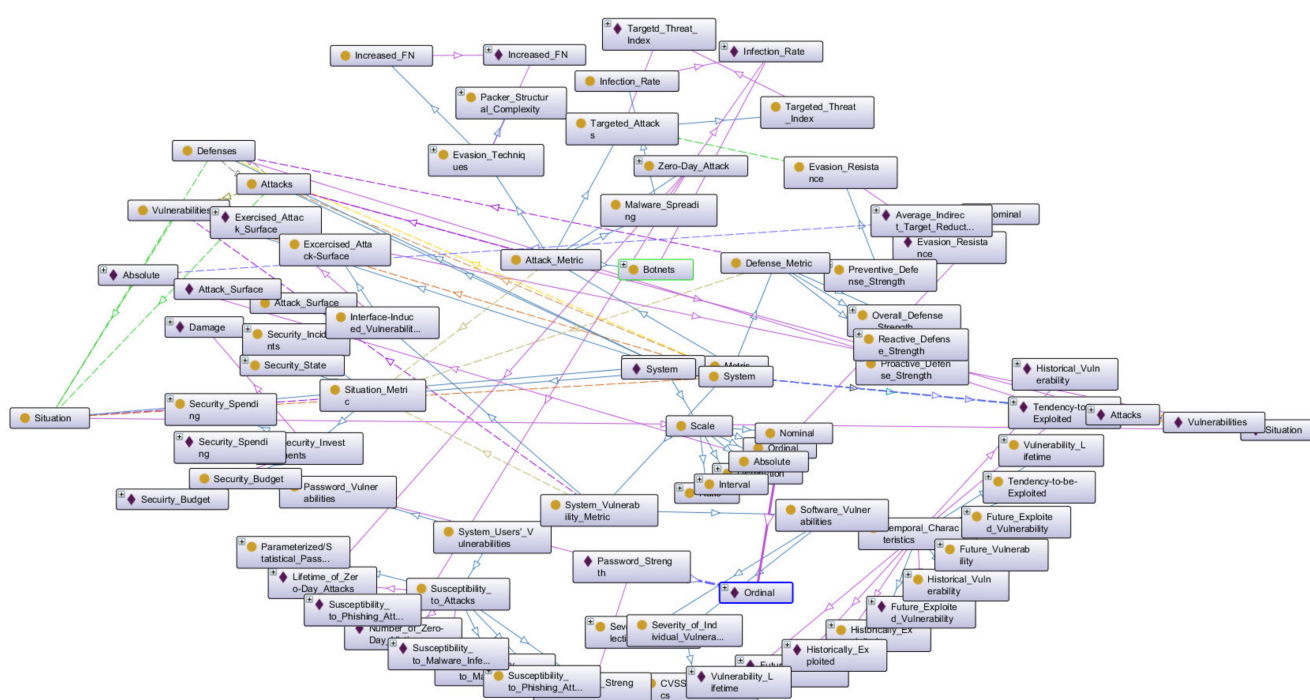


Figure 3. Classification of attack classes in security context ontology.

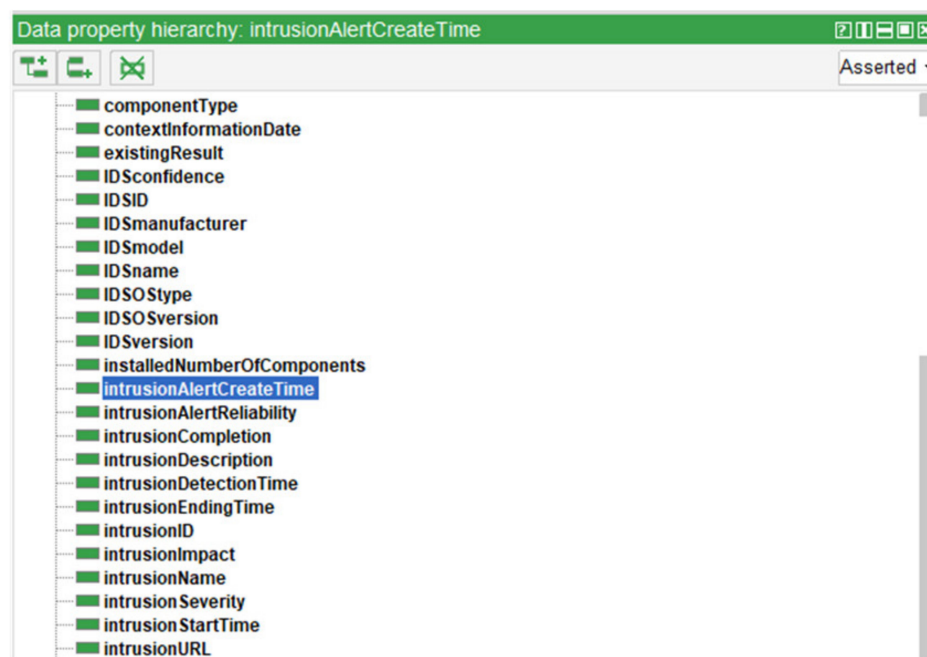


Figure 4. Data properties of *IntrusionDetectionSystem* class.

Clear information can be represented only when the components and their relationships are defined. Therefore, the relationships between components were represented by defining them with the OWL property. In other words, the OWL property implies a relationship, and the property representation is categorized into object and data type properties, in which the object properties represent relationships between two individuals, and the data type properties represent relationships between individuals and data values. Semantic web rule language (SWRL) is used for rule-based inference in the relationship between classes, properties, and individuals using ontology for the security context recognition service of the system. SWRL defines rules for new knowledge to be obtained through the relationship

of information in ontology, and the Jess engine is used to execute the defined SWRL. The inference rule was defined as shown in Table 2 by referring to the security context ontology to infer the rules for security context awareness.

Table 2. Example of ontology inference rule (semantic web rule language, SWRL) for security context awareness.

Inference Attack Target	Ontology Inference Rule
Denial of service (DoS) attack detection	DoS(?threatdos), NetworkContext(?netcontext), SystemContext(?syscontext), receivedFormattedIntrusion(?IRS, ?intrusion), contextInformationDate(?netcontext, ?netdate), contextInformationDate(?syscontext, ?sysdate), networkAnomaly(?netcontext, ?netan), equal(?sysdate, ?intdate), greaterThan(?netan, 7), greaterThan(?syslatency, 7) -> indicates(?syscontext, ?threatdos)
Backdoor attack detection	BackDoors(?threatbackdoors), SystemContext(?syscontext), contextInformationDate(?syscontext, ?sysdate), intrusionDetectionTime(?intrusion, ?intdate), systemNumberOfUsersLogged(?syscontext, ?sysUsers), equal(?sysdate, ?intdate), greaterThan(?sysUsers, 3), greaterThan(?sysprocesses, 3) -> indicates(?syscontext, ?threatbackdoors)

The security context inference rule using the security context ontology infers the optimal response to the security intrusion of the power system. The inference engine executes an inference process to determine the optimal response to a specific intrusion by considering the ontology instance that reflects the information regarding the security policy, response, security context, and intrusion alert defined by the system administrator. The security context inference module calls the evaluation module for attack detection, updates the security context information inferred from the security context ontology using the Jena library, and updates the new situation result processed in the response layer.

4. Experiment and Evaluation

The precision of system access authorization and the attack response precision of the intelligent integrated access control system can determine the inferring ability, which can suggest the optimal countermeasure in the intelligent integrated access control system when a security attack occurs, based on the access control inference rules for the evaluation of intelligent access control proposed herein. Precision and recall were used to evaluate the response to attacks. Precision refers to the percentage of the actual attack situation among the results predicted by the attack situation; it is defined as shown in Equation (1).

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

Recall refers to the ratio of accurately predicting the attack situation out of the actual attack situation using the proposed method; it is defined as shown in Equation (2).

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

4.1. Dataset

A simulation was conducted to assess whether the overall security context can be determined when various security events occur by generating attack events using 200 pieces of security context information. Attack events were collected based on behavior pattern data obtained during the execution of malicious codes and normal files based on Win32 application programming interface (API) behavior information. The Win32 API behavior patterns were collected when a specific event occurred by executing 500 malicious codes

generated by a malicious code generator and 400 normal files. In the set of API behavior pattern sequences, 70% are normal behavior patterns, and the remaining 30% are malicious behavior patterns. In the API behavior pattern collection process, five datasets were created as shown in Table 3, with the ratio of malicious behaviors being 0.1, 0.2, 0.3, 0.4, and 0.5, respectively. If the security system detects malicious code using the generated experimental dataset, it is compared with the detection result using the inference rule proposed in this paper.

Table 3. Experimental dataset.

Dataset	Normal Behavior Sequence Set Size	Malicious Behavior Sequence Set Size	Malware Inclusion Rate	Total Number of Records
Dataset 1	5000	71	0.1	6211
Dataset 2	5000	71	0.2	7340
Dataset 3	5000	71	0.3	8178
Dataset 4	5000	71	0.4	9981
Dataset 5	5000	71	0.5	10,143

If a process corresponding to the behavior pattern is discovered within the system based on the basic elements of the behavior pattern performed by each process using the API calling function, then the host ID, user ID that executed the process, process ID, event occurrence time, execution file (or process) name, and behavior pattern element are collected and saved, as shown in Table 4.

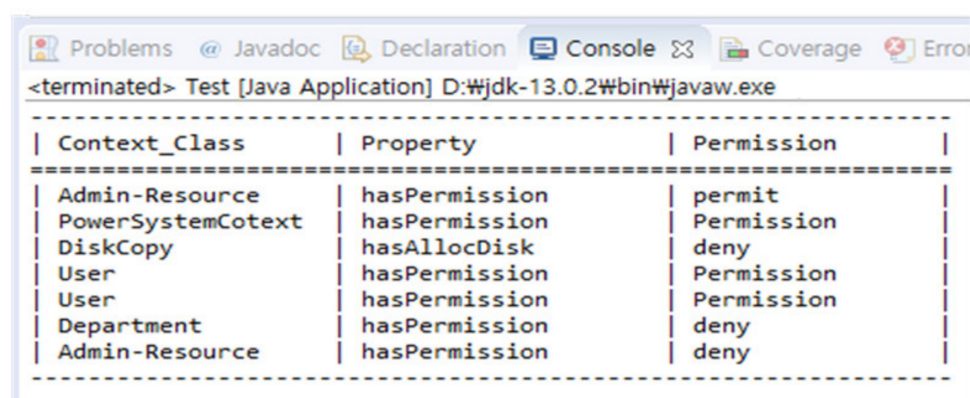
Table 4. Examples of process collection information.

Index	HostID	UserID	PID	Time	Filename	Feature Index	Result	Parameter
23112	1	admin	3145	26-12-2020 14:04:55,761	chrome.exe	2(OpenThread)	SUCCESS	Thread ID: 8715
23231	1	admin	3271	26-12-2020 14:05:03,312	chrome.exe	6(OpenProcess)	SUCCESS	
...
134121	1	admin	3419	26-12-2020 14:11:43,651	iexplore.exe	8(CreateProcess)	SUCCESS	
138613	1	admin	3581	26-12-2020 14:11:44,134	iexplore.exe	4(CreateProcess)	SUCCESS	

Through this process, attack events such as backdoor, password stealing, and denial-of-service attacks were generated, and the inference rules defined for each security context were expanded and tested.

4.2. Precision of Security Context Ontology-Based Access Control Inference

In this section, in order to verify the accuracy of the security context ontology-based access control inference, when malicious code is detected, the access authority setting of the proposed access control system is tested. Authorization management is necessitated in the intelligent access control model owing to personal information leakage. Thus, the proposed access control model was applied through the request authorization setting for each internal level. The system resource authorization information was analyzed by inferring the relationship setting of the security context access control ontology, as shown in Figure 5.



Context_Class	Property	Permission
Admin-Resource	hasPermission	permit
PowerSystemCotext	hasPermission	Permission
DiskCopy	hasAllocDisk	deny
User	hasPermission	Permission
User	hasPermission	Permission
Department	hasPermission	deny
Admin-Resource	hasPermission	deny

Figure 5. Results of applying inference rules.

The result of setting the correct resource authorization based on the malicious code detection result that applied the proposed inference-based intelligent access control model is shown in Table 5. In the experiment, after executing the malicious code sample performing the information leakage according to the number of malicious activities, the malicious code scanner analysis tool was used to detect the malicious behavior. When a malicious activity is detected, the access control module infers the security context of the malicious activity and sets appropriate access authorization. When 50 malicious actions were performed by malicious code, the access permission setting rate inferred from the access control module was 82%, but as the number of malicious actions increased, the performance of the access permission setting rate improved. This means that as the number of security attacks increases, the rate of setting the correct access rights increases. By detecting the information leakage behavior when detecting malicious codes, it was discovered that appropriate access was authorized through security context access control ontology inference for each security context. The correct authorization rate using access control inference was 87% on average when a malicious code was detected.

Table 5. Results of access authorization of intelligent access control when malicious behavior is detected.

Number of Malicious Behaviors	Malware Code Scanner Analysis Tool	Detection Rate	Correct Access Authorization Rate
Case 50	42	74%	82%
Case 100	83	83%	83%
Case 500	437	87.4%	89.4%
Case 1000	873	87.3%	91.7%

The precision of the security attack detection inference of the security context awareness service was measured via simulated hacking. The precision, recall, and F-measure of attack detection were evaluated using 200 pieces of random security context information to verify the precision of the attack detection inference through the security context ontology and inference engine.

The purpose was to obtain the precision, recall, and F-measure through the security context query for the measurement, and the inference result of each input query was compared with the inference value to be provided. The types of queries and the number of inference rules used are as shown in Table 6.

Table 6. SWRL query for attack detection by security context.

Security Context	Query	Number of Inference Rules
Backdoor attack detection	BackDoors(?threatbackdoors), SystemContext(?syscontext), contextInformationDate(?syscontext, ?sysdate), intrusionDetectionTime(?intrusion, ?intdate), systemNumberOfUsersLogged(?syscontext, ?sysUsers), equal(?sysdate, ?intdate), greaterThan(?sysUsers, 3), greaterThan(?sysprocesses, 3) -> indicates(?syscontext, ?threatbackdoors)	4
Password stealing attack detection	PasswordAttacks(?threatpassword), SystemContext(?syscontext), contextInformationDate(?syscontext, ?sysdate), intrusionDetectionTime(?intrusion, ?intdate), systemCPUUsage(?syscontext, ?syscpu), systemLatency(?syscontext, ?syslatency), equal(?sysdate, ?intdate), greaterThan(?syscpu, ?4), greaterThan(?syslatency, 4) -> indicates(?syscontext, ?threatpassword)	10
Similar behavior intrusion detection	Result(?result), hasIntrusionType(?intrusion, ?threat), hasIntrusionType(?intrusionres, ?threatres), hasTarget(?intrusion, ?target), relatedIntrusion(?result, ?intrusionres), assetLevelOfImportance(?targetres, ?targetresimp), responseStatus(?intrusion, ?intstatus), SameAs (?threat, ?threatres) -> hasSimilar-Result(?intrusion, ?result)	15

The experimental results of the attack detection query by security context are shown in Table 7. TP is a value that is correctly inferred among the inferred results, FN is a non-inferred value, and FP is an incorrectly inferred value among the inferred values.

Table 7. Experimental results of attack detection by security context (1).

Security Context	Number of Inferences	TP	FN	FP
Backdoor attack detection	797	165	182	313
Password stealing attack detection	454	287	87	78
Similar behavior intrusion detection	243	221	6	9

Precision, recall, and F-measure, which is the sum of precision and recall, were used to evaluate detection rate. Precision represents the ratio of distinguished values among the total inferred values; recall represents the rate of distinguished values among all data of all groups. F-measure represents the reliability of the inference result and is defined as shown in Equation (3).

$$F - measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3)$$

The results of comparing and evaluating the precision, recall, and F-measure of each security context information record using Equation (3) are shown in Table 8. The attack detection rate for each security context was 70% on average.

Table 8. Experimental results of attack detection by security context (2).

Security Context	Precision	Recall	F-Measure
Backdoor attack detection	0.3452	0.47550	0.40000
Password stealing attack detection	0.78630	0.76738	0.77673
Similar behavior intrusion detection	0.96087	0.97357	0.96718

4.3. Attack Response Rate of Intelligent Integrated Access Control

The attack response rate of the intelligent integrated access control system was measured in the same manner as the attack detection inference precision of the security context awareness service. The types of queries and number of inference rules used are shown in Table 9.

Table 9. SWRL query of intelligent integrated access control system for attack response.

Security Context	Query	Number of Inference Rules
Intrusion warning reliability	NotThreat(?threatcon), SystemContext(?syscontext), hasIntrusionType(?intrusion, ?threat), indicates(?syscontext, ?threatcon), protects(?response, ?sgres), threatens(?threat, ?sgint), IDSconfidence(?ids, ?idsconf), contextInformationDate(?syscontext, ?sysdate), intrusionDetectionTime(?intrusion, ?intdate), responseStatus(?intrusion, ?status), equal(?idsconf, "high"), equal(?status, "Pending"), equal(?sysdate, ?intdate), SameAs (?sgres, ?sgint) -> recommendedResponses(?intrusion, ?response), intrusionAlertReliability(?intrusion, "medium")	8
Optimal response to intrusion detection	hasarget(?intrusion, ?target), potentialOptimumResponses(?intrusion, ?respon), receivedFormattedIntrusion(?irs, ?intrusion), numberOfPotentialOptimumResponses(?intrusion, ?numOpResp), responseCost(?respon1, ?respCost1), responseCost(?respon2, ?respCost2), equal(?aloi, "low"), greaterThan(?numOpResp, 1) -> optimumResponse(?intrusion, ?respon1)	14
Recommended inference for results	hasSimilarResult(?intrusion, ?result), responseResult(?result, ?resresult), responseStatus(?intrusion, ?respStatus), equal(?respStatus, "Pending"), equal(?resresult, "Satisfactory") -> neededRecommendedInference(?intrusion, false)	6

The experimental results of the attack response query of the intelligent integrated access control system are shown in Table 10. TP is a correctly inferred value among the inferred results, FN is a non-inferred value, and FP is an incorrectly inferred value among the inferred values.

Table 10. Experimental results of attack response of intelligent integrated access control system (1).

Security Context	Number of Inferences	TP	FN	FP
Intrusion warning reliability	165	76	33	56
Optimal response to intrusion detection	89	54	18	17
Recommended inference for results	65	45	12	8

The results of comparing and evaluating the precision, recall, and F-measure for each security context for the security context information records obtained using Equation (3) are shown in Table 11. The average attack response rate was 72.8%.

Table 11. Experimental results of attack response (2).

Security Context	Precision	Recall	F-Measure
Intrusion warning reliability	0.5758	0.6972	0.6307
Optimal response to intrusion detection	0.7606	0.7500	0.7552
Recommended inference for results	0.8491	0.7895	0.8182

5. Conclusions

Recently, many industrial security attacks, including advanced persistent threat (APT) attacks, not only target specific systems, but also have a wide attack range and cause significant damage. Countermeasures against such large-scale attacks necessitate a technology that can determine the security context type by analyzing security events occurring in numerous security systems, e.g., firewalls, vaccines, intrusion detection systems, and intrusion prevention systems within the power system. Although such a comprehensive determination is currently performed by system administrators and security experts, intelligent integrated security technology is necessitated to achieve rapid responses. Therefore, in this paper, we proposed an intelligent access control framework that can recognize the security context based on ontology inference for the security management of the power system. The intelligent access control framework was designed as a framework that enables collaboration within the smart grid environment, and a system administrator was designed to transmit access control policy information required between the power service principal and the agent. In this regard, a security context ontology was designed, through which the inference rules were defined.

For the verification of the proposed framework, a simulation was conducted to generate an attack event using 200 security context information records. Through experiments, we measured the security context access control ontology-based access control inference, the attack detection inference of the security context recognition service, and the attack response rate of the intelligent integrated access control system. As a result of the experiment, the attack detection rate of security context ontology-based access control was evaluated as 70%, and the attack response rate of intelligent integrated access control was evaluated as 72.8%.

Author Contributions: Conceptualization, H.K. and J.C.; methodology, H.K. and J.C.; investigation, J.C.; resources, H.K.; writing—original draft preparation, H.K. and J.C.; writing—review and editing, H.K. and J.C.; visualization, J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Korea Electric Power Corporation (grant number: R18XA06-12).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

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

References

1. Kamienski, C.A.; Borelli, F.F.; Biondi, G.O.; Pinheiro, I.; Zyrianoff, I.D.; Jentsch, M. Context Design and Tracking for IoT-Based Energy Management in Smart Cities. *IEEE Internet Things J.* **2018**, *5*, 687–695. [\[CrossRef\]](#)
2. Figueroa-Lorenzo, S.; Anorga, J.; Arrizabalaga, S. A Role-Based Access Control Model in Modbus SCADA Systems. A Centralized Model Approach. *Sensors* **2019**, *19*, 4455. [\[CrossRef\]](#) [\[PubMed\]](#)
3. Bertin, E.; Hussein, D.; Sengul, C.; Frey, V. Access control in the Internet of Things: A survey of existing approaches and open research questions. *Ann. Telecommun.* **2019**, *74*, 375–388. [\[CrossRef\]](#)
4. Bettini, C.; Brdiczka, O.; Henriksen, K.; Indulska, J.; Nicklas, D.; Ranganathan, A.; Riboni, D. A survey of context modelling and reasoning techniques. *Pervasive Mob. Comput.* **2010**, *6*, 161–180. [\[CrossRef\]](#)

5. Ghazal, R.; Malik, A.K.; Qadeer, N.; Raza, B.; Shahid, A.R.; Alquhayz, H. Intelligent Role-Based Access Control Model and Framework Using Semantic Business Roles in Multi-Domain Environments. *IEEE Access* **2020**, *8*, 12253–12267. [[CrossRef](#)]
6. Zhang, R.; Giunchiglia, F.; Crispo, B.; Song, L.Y. Relation-Based Access Control: An Access Control Model for Context-Aware Computing Environment. *Wirel. Pers. Commun.* **2010**, *55*, 5–17. [[CrossRef](#)]
7. Kayes, A.S.M.; Rahayu, W.; Dillon, T.; Chang, E.; Han, J. Context-aware access control with imprecise context characterization for cloud-based data resources. *Future Gener. Comput. Syst. Int. J. Escience* **2019**, *93*, 237–255. [[CrossRef](#)]
8. Yang, Y.C.; Wu, L.F.; Yin, G.S.; Li, L.J.; Zhao, H.B. A Survey on Security and Privacy Issues in Internet-of-Things. *IEEE Internet Things J.* **2017**, *4*, 1250–1258. [[CrossRef](#)]
9. Xu, G.Q.; Cao, Y.; Ren, Y.Y.; Li, X.H.; Feng, Z.Y. Network Security Situation Awareness Based on Semantic Ontology and User-Defined Rules for Internet of Things. *IEEE Access* **2017**, *5*, 21046–21056. [[CrossRef](#)]
10. Sharma, V.; Choudhary, G.; Ko, Y.; You, I. Behavior and Vulnerability Assessment of Drones-Enabled Industrial Internet of Things (IIoT). *IEEE Access* **2018**, *6*, 43368–43383. [[CrossRef](#)]
11. Zhou, J.; Cao, Z.F.; Dong, X.L.; Vasilakos, A.V. Security and Privacy for Cloud-Based IoT: Challenges, Countermeasures, and Future Directions. *IEEE Commun. Mag.* **2017**, *55*, 26–33. [[CrossRef](#)]
12. Kirrane, S.; Mileo, A.; Decker, S. Access Control and the Resource Description Framework: A Survey. *Semant. Web* **2017**, *8*, 1–42. [[CrossRef](#)]
13. Ouaddah, A.; Mousannif, H.; Abou Elkalam, A.; Ouahman, A.A. Access control in the Internet of Things: Big challenges and new opportunities. *Comput. Netw.* **2017**, *112*, 237–262. [[CrossRef](#)]
14. Uddin, M.; Islam, S.; Al-Nemrat, A. A Dynamic Access Control Model Using Authorising Workflow and Task-Role-Based Access Control. *IEEE Access* **2019**, *7*, 166676–166689. [[CrossRef](#)]
15. Kayes, A.S.M.; Han, J.; Colman, A. OntCAAC: An Ontology-Based Approach to Context-Aware Access Control for Software Services. *Comput. J.* **2015**, *58*, 3000–3034. [[CrossRef](#)]
16. Servos, D.; Osborn, S.L. Current Research and Open Problems in Attribute-Based Access Control. *Acm Comput. Surv.* **2017**, *49*, 65. [[CrossRef](#)]
17. Choi, C.; Esposito, C.; Wang, H.X.; Liu, Z.; Choi, J. Intelligent Power Equipment Management Based on Distributed Context-Aware Inference in Smart Cities. *IEEE Commun. Mag.* **2018**, *56*, 212–217. [[CrossRef](#)]
18. Choi, C.; Choi, J. Ontology-Based Security Context Reasoning for Power IoT-Cloud Security Service. *IEEE Access* **2019**, *7*, 110510–110517. [[CrossRef](#)]
19. Mozzaquatro, B.A.; Agostinho, C.; Goncalves, D.; Martins, J.; Jardim-Goncalves, R. An Ontology-Based Cybersecurity Framework for the Internet of Things. *Sensors* **2018**, *18*, 3053. [[CrossRef](#)]