Building Confidence using Beliefs and Arguments in Security Class Evaluations for IoT

Manish Shrestha
eSmart Systems AS
Halden, Norway
manish.shrestha@esmartsystems.com

Christian Johansen
Department of Technology Systems
University of Oslo
Oslo, Norway
cristi@ifi.uio.no

Josef Noll
Department of Technology Systems
University of Oslo
Oslo, Norway
josef.noll@its.uio.no

***Abstract*—The proliferation of IoT (Internet of Things) though making life easier, comes with security and privacy challenges. We have previously proposed a security classification methodology meant to help in practice build IoT systems focused on security during the development process. This method departs from classical risk analysis and certification methods in two ways: (i) it can be used at design time and (ii) it caters for the needs of system designers by helping them to identify protection mechanisms necessary for the connectivity required by their system under development. However, similarly to many risk analysis methods, this methodology was unable to provide assurance in the evaluation results. In this paper, we add two confidence parameters: *belief* and *uncertainty* to the assessment tree of arguments of a class. Thus, the final result is now a tuple <*C, B, U*>, where *C* is the class to which the system belongs, together with a belief measure *B* in the evaluation aspects of *C*, and the uncertainty *U* in the evaluation details. Looking at the confidence parameters tells how well the security assessment is justified. To exemplify this enhanced security classification methodology, we systematically apply it to control mechanisms for Smart Home Energy Management Systems.**

***Index Terms*—Security Classification, Security assurance, Uncertainty, Confidence, Security labelling**

I. INTRODUCTION

Internet of Things (IoT) is widely adopted in major sectors including critical infrastructures such as smart grids and privacy-sensitive domains such as smart homes. Because IoT devices produce sensitive data and have limited memory and processing power, IoT systems are easy targets for launching cyber attacks. Despite the efforts to secure IoT systems, attacks are increasing[1]. One of the reasons behind this is the lack of security awareness in end-users preferring cheaper and easy to install insecure products. Traditional certification approaches s.a. Common Criteria are usually expensive and take more than a year to get certified [2]. Investing in such certification does not pay off because of the lower cost and short life span of IoT products. Even if we start to see standards for cybersecurity for consumer IoT, s.a. the recent ETSI TS 103 645, a framework for designing and evaluating IoT systems for an appropriate level of security still does not exist.

This work is funded by eSmart Systems AS and the Research Council of Norway through the project IoTSec - "Security in IoT for Smart Grids", with number 248113/O70. A long version of this paper is available as [1].

[1]https://www.forbes.com/sites/zakdoffman/2019/09/14/dangerous-cyberattacks-on-iot-devices-up-300-in-2019-now-rampant-report-claims/

We have previously proposed a notion of security classes [3] to address the aforementioned challenges. By systematically applying our security classification methodology, a system designer (or user or certification body, for the same matter) can classify the security of their system on a scale from A to F where A represents the best security level. Most methodologies for security classification or risk analysis are based on knowledge and experience of security experts executing the evaluations. For our target audience, i.e., end-users or system designers/developers, only claiming a security class without justification is insufficient.

To build trust in the security classification one needs to answer questions like: "How confident are the experts in their result?" or "Were any decisions made under uncertainty?". In response, we introduce in this paper two new parameters in the security classification methodology, namely *belief* and *uncertainty*, described in section III after briefly recalling our previous security classification methodology in Section II. In Section IV the enhanced security classification methodology is applied to a Smart Home Energy Management System (SHEMS), ending up with a comparative discussion.

II. SECURITY CLASSIFICATION METHODOLOGY

We have proposed in [3] a security classification methodology, which extends the ANSSI classification, for analysing and evaluating the security of complex connected systems. This methodology is built around three main factors (see Figure 1): Connectivity, Security mechanisms, and Impacts. Connectivity reflects how the system is exposed to attacks, whereas security mechanisms evaluate the security features protecting the system (Protection Level). Connectivity and Protection level combined form the Exposure.

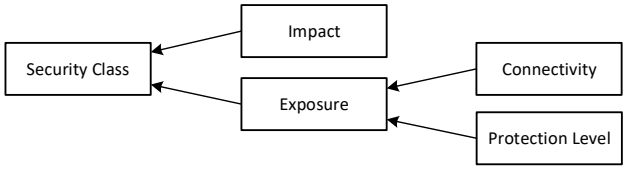

Fig. 1: Process of computing a security class.

We have considered five levels of connectivity (C):

- **C1** : Includes completely closed/isolated systems.
- **C2** : Includes the system with wired Local Area Network and does not permit operations from outside the network.
- **C3** : Includes C2 systems that use wireless technologies.
- **C4** : Includes systems with private or leased infrastructure, which may permit remote operations (e.g., VPN, private APN, etc).
- **C5** : Includes distributed systems with public infrastructure, i.e., like the C4 category only that the communication infrastructure is public.

Similarly, there are five protection levels (P), reflecting the security mechanisms in the system. To determine the protection level, relevant security criteria are defined, and for each criterion, the respective security mechanisms are derived. The security mechanisms are then grouped to form individual protection levels where a higher protection level includes all the security functionalities of lower protection levels, plus additional functionalities. Protection level P1 represents no security mechanisms whereas the protection level P5 represents the strongest protection mechanisms. The evaluation of protection mechanisms is conducted by security experts. Table I guides the evaluation of exposure from connectivity and protection levels.

TABLE I: Calculation of Exposure Levels.

P1	E4	E4	E5	E5	E5
P2	E3	E4	E4	E5	E5
P3	E2	E3	E3	E4	E4
P4	E1	E1	E2	E2	E3
P5	E1	E1	E1	E1	E2
Protection/ Connectivity	C1	C2	C3	C4	C5

TABLE II: Calculation of Security Classes.

Catastrophic	A	C	E	F	F
Major	A	B	D	E	F
Moderate	A	B	C	E	E
Minor	A	A	B	D	D
Insignificant	A	A	A	C	C
Impact/ Exposure	E1	E2	E3	E4	E5

The impacts also have five levels taken from ANSSI: Insignificant, Minor, Moderate, Major, and Catastrophic. A security class is determined using impact and exposure according to the look-up Table II.

In a typical SHEMS, devices are remotely controlled (hence, connectivity is C5) and the control data are well encrypted and monitored (hence, protection level P4), and so the acquired Exposure would be E3 (cf. Table I). Given that a compromised SHEMS is seen as a major impact, the final security class would be "D". Details can be found in [3], [4].

Table II shows variations on these calculations, e.g., exposures E1, E2, or E3 with impact "catastrophic" result in classes A, C, resp. E. However, there are no more explanations than this look-up table, whereas the details of choosing protection mechanisms and connectivity are considered expert judgement art. Hence, we see the need to introduce confidence in the analysis and arguments to justify the results and quantify the (un)certainty with which the decision is made.

III. CONFIDENCE IN A SECURITY CLASS

The main contribution of this paper is to enhance the security classification method with the ability to argue and reflect the level of confidence for each decision. By *confidence*, we mean the degree to which one agrees on the result of the assessment (belief) and the degree to which the expert lacks knowledge about the assessment (uncertainty). To enrich the security classification method, we propose to represent the assessment result using a three tuple $<C, B, U>$, e.g., the evaluation $<A, 84, 16>$ means that the result is class A with 84% confidence and 16% uncertainty. The 84% belief is meant to say that we have high confidence in the coverage of all necessary security measures to justify the protection (P), exposure (E), and security class. An uncertainty of 16% would indicate that there is a moderate lack for justification of some of the arguments.

A. Assessment of Belief and Uncertainty

To understand the concept of belief, let us consider a wireless sensor network where an expert makes a claim C1: "Source node adequately encrypts data before sending to the destination". An expert may justify this claim by referring to the technical documentation from the vendor claiming that data is encrypted during transfer. If the vendor is reliable one may set higher belief on the claim C1, say 90%. However, there may be some errors during design or implementation which may result in unencrypted data. So, the remaining 10% represents the uncertainty of the claim. If one can experimentally verify the C1, e.g. through a penetration testing tool, C1 could be fully trustworthy, which means 100% belief. This 100% is called a plausible belief or plausibility. Hence, plausibility is the maximum belief that can be obtained if all the evidence is provided. In another case where an exploitable flaw is discovered in an encryption algorithm, then disbelief in the claim may arise. Let us say that the estimated disbelief is 30%, then the highest level of belief that one can make in this situation is 70% (reduced plausibility).

One of the widely used approaches to quantifying belief and uncertainty is the Dempster-Shafer theory, which is a generalization of probability theory that allows representing incomplete knowledge by the notion of upper and lower probabilities (belief and plausibility) [5]. Belief (Bel) represents the strength of the existing pieces of evidence that support a given statement. Similarly, Plausibility (Pl) is the upper bound on the belief that could be obtained by adding the evidence to support the statement. Thus, belief is less than or equal to plausibility ($0 \leq Bel \leq Pl \leq 1$).

Uncertainty is the degree of lack of knowledge or evidence to justify the claim. It can be calculated as the difference between plausibility and belief on the acceptance of the claim. Additionally, $Pl < 1$ indicates the existence of disbelief meaning that there is some evidence against the claim. In our context, we reuse the definition of belief and plausibility

from the Dempster-Shafer theory. After belief and plausibility evaluation, uncertainty can be calculated as:

$$Uncertainty = Plausibility - Belief \quad (1)$$

B. Specifying security arguments

In a security class evaluation, a series of security arguments are made. Govier defines an argument as "a set of claims in which one or more of them —the premises— are put forward so as to offer reasons for another claim, the conclusion" [6]. To demonstrate an argument in our case, let us take an example of the assessment of the physical security of an IoT device. During the assessment, it was found that the device is located inside the apartment, and is physically accessible only by the residents. Moreover, the device has a tamper detection mechanism which notifies about unauthorized tampering. Therefore, it can be concluded that the IoT device has adequate physical security. In this example, there is a *set of claims and reasons*: IoT device has a secure location because it is installed inside the apartment; IoT device notifies about tamper activities because it has tamper detection mechanism; IoT device has good physical security because it is placed in a safe place and the owner gets notification about device tampering.

The confidence in the conclusion "IoT device has good physical security" depends on the confidence in the reasons presented. The amount of trust in the *ground of the claim* also impacts the confidence. In our example, the ground of the argument is: "if the physical location is secure and a tamper detection functionality exists, the device is physically secure". If the ground is weak, the trust in the conclusion would also be weak. This implies that the amount of confidence in the conclusion depends on the trust in the main claim and the supporting components of the claim.

Properly structured arguments with appropriate justifications and evidence make the expert opinion explicit, resulting in improved communication between experts. This can also help identify missing evidence and poor assumptions. Structured arguments are widely used for justification of decisions, e.g., in safety cases [7], assurance cases [8], [9] and trust cases [10]. In structured cases, the main conclusion is backed up by the evidence. To make the arguments clearer, there are various methods and notations such as Goal Structuring Notation (GSN) [11], Claim Argument Evidence (CAE) [7], or Toulmin argument model [12], that can help experts to structure their arguments. All the above methods represent the argument as a tree structure where the root node is the main claim which further grows into child nodes that provide the justifications using sub-claims and evidence.

Our security classification methodology involves a series of systematic steps to achieve a final security class. We here propose to structure such an assessment as an argumentation model. Structured arguments provide the reasons to support the security claims. These reasons can be seen as security requirements for the assessment, and also guide security experts to determine to which degree the requirements that are fulfilled is reflected by the confidence (belief and uncertainty). Fig. 2 shows an example of the security class evaluation step as an argumentation model using GSN. After the assessment is structured as the argumentation model, the weights, beliefs and plausibility are assigned to the claims and evidence produced.

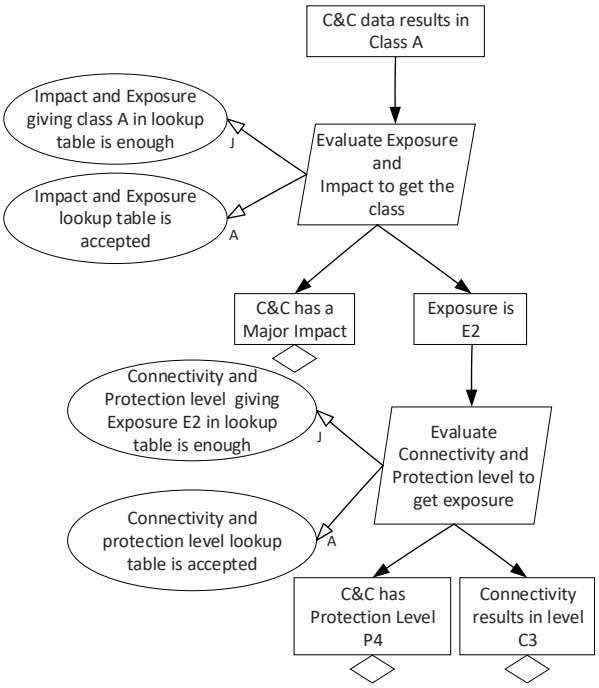

Fig. 2: Class A evaluation using Goal Structuring Notation. (In the figure "J" and "A" point to the justification and assumption made to support the strategy represented by a parallelogram. The rectangle represents the claim and the diamond symbol represents that the digram is incomplete and should be expanded further.)

C. Aggregation of confidence parameters

The result of a security class assessment is represented by the class label with the overall confidence parameter (belief and uncertainty). Thus, after specifying confidence parameters to the security arguments, the result is aggregated to represent an overall assessment. Various aggregation methods have been proposed for structured cases, e.g., Wang et. al. [13] proposed generalized confidence propagation rules for safety cases based on Dempster-Shafer theory. In their D-arg rule, the aggregated belief is calculated as a weighted mean. However, weighted means are not sensitive to extreme lower values of beliefs. Similarly, in their FC-Arg rule, the aggregated belief is the result of multiplying the individual beliefs [13]. The result of such beliefs would always be diminished if we add more evidence that has belief less than 1. However, normal intuition is that, with an increase of evidence, the beliefs should strengthen. Aggregation rules in trust cases have similar problems, e.g., see aggregation rules C-arg, SC-arg and NSC-arg in [10]. Noll et. al., in their Multi-Metrics (MM) approach [14], have claimed that quadratic functions reflect the aggregation better than linear approaches.

In our case, the arguments we have considered contribute individually to the overall goal. Based on the significance of

each component in the system's security, we assign appropriate weights in the range [0-100]. We then compare the weighted mean approach with MM approach for calculating beliefs.

1) Weighted mean approach: The aggregated belief using weighted mean for beliefs (b) and weights (w) can be calculated using the formula:

$$AggregatedBelief(c) = \frac{\sum_i^n w_i b_i}{\sum_i^n w_i} \tag{2}$$

2) MM approach: The MM approach uses the Root Mean Squared Weighted Data (RMSWD) to aggregate criticality values and is expressed as:

$$X = \sqrt{\sum_i \left(\frac{x_i^2 W_i}{\sum_i^n W_i} \right)} \tag{3}$$

where X is the aggregated criticality, x_i is the criticality of i^{th} component, and W_i is calculated from the component weight w_i as:

$$W_i = \left(\frac{w_i}{100} \right)^2 \tag{4}$$

In the original work, criticality x_i is defined as the complement of security, privacy or dependability metrics [14]. In our context, we use the complement of belief value (*100 - belief*) to express criticality. Finally, the aggregated belief (Bel) is computed as a complement of X (i.e., $Bel = 100 - X$). Thus, using equation 3, Bel can be expressed as:

$$Bel = 100 - \sqrt{\sum_i \left(\frac{(100 - bel_i)^2 W_i}{\sum_i^n W_i} \right)} \tag{5}$$

where bel_i is the individual belief value of the component under consideration.

D. Underlying principles for aggregation

Belief aggregation depends on how the arguments are presented. There are cases when there are multiple justifications independent of each other fulfilling the same claim, or sometimes each justification contributes towards the fulfillment of the claim to some extent. Here we describe the principles to guide the aggregation mechanism in special cases.

1) **Maximum belief**: If justifications are overlapping and one justification includes another, the highest belief should be considered. For example, to justify the claim "Data is encrypted", there are two evidences with different beliefs: 1) Document from the vendor describing that the data is encrypted (belief = 90%), and 2) Experimental verification for encryption (belief = 100%). In this case, we simply select the highest belief because the information from the vendor's document is subsumed by the experimental verification. Thus,

   $Aggregated\ Belief = Max(b1, b2)$

   where $b1$ and $b2$ are beliefs on overlapping claims where justification of one of the claim includes the other.
2) **Zero belief**: If the belief for any of the claims in the evaluation for protection level is zero, then the total belief should be zero because the class is determined based on the previously specified requirements of security functionality (*sf*). If one of the functionality has no belief at all, then the whole claim for that protection level fails and it must be evaluated against the lower protection level. The same applies to the aggregation of protection criteria (c) towards the protection level.

   if $c.securityFunctionalities.Any(sf.belief = 0)$ **then**
   $c.belief = 0$
   end if
3) **Minimum belief**: Typically, it is assumed that the impacts and connectivity have full beliefs; otherwise, if the beliefs are lower than 100%, the resulting aggregated belief should be the lowest one. For example, if the exposure is E2, with belief 90% and the Impact is Major with 60% then the class obtained should have belief 60% instead of the average. This is because both of them are equally important and required for evaluation. Thus, the averaged belief has no meaning. Hence,

   $Aggregated\ Belief = Min(b1, b2)$.

IV. CASE STUDY

To demonstrate the applicability of confidence in security classifications, we have selected a use case involving command and control in SHEMS. The scenarios for the use case are built upon our previous work [4] which used two principal methods to control the IoT devices: centralized and edge control. The centralized control has higher connectivity and major impacts, therefore resulting in class D; whereas, in the edge control scenario, the connectivity is reduced, while also reducing the impacts, thus resulting in class A. We continue here to look at the edge control scenario and follow Section III to add confidence reasoning.

A. Protection level evaluation

For an IoT device control system, we considered Data Encryption (e.g., for securing control commands), Access control, and Monitoring & Analysis, as relevant criteria. We first analyze the security mechanisms available for each of these security criteria, in order to determine the protection level following the summary in Table III. We assume that the answers for the existing security functionalities in the C&C component fall onto the column (i.e., protection level) P4, i.e., two functionalities are not present. Next, we discuss confidence parameters for each protection criteria and their mechanisms.

1) Data Encryption: The following sub-claims were considered to satisfy the P4 level requirements:

- **C&C data is encrypted between IoT hub and devices**: The belief on this claim is 100% and is justified by the vendor's document and lab test.
- **Data encryption uses a strong encryption algorithm**: It has been verified that data is encrypted with AES 128-bit encryption which is considered strong for home network. Thus, the belief in this sub-claim is also set to 100%.

TABLE III: Protection Level Requirements for C&C in a SHEMS application.

Protection Criteria	Security Functionality	P5	P4	P3	P2	Table IV
Data Encryption	C&C data is encrypted between IoT hub and devices	✓	✓	✓	✓	✓
	Data encryption uses a strong encryption algorithm	✓	✓	✓		✓
	End-to-end encryption is supported	✓	✓			✓
	Does not use custom encryption algorithms	✓	✓			✓
Access Control	Disable remote access functionality	✓				
	Weak and default credentials are not allowed	✓	✓	✓		✓
	Enable Multi-factor Authentication	✓	✓			N/A
Monitoring and Analysis	Monitor system components	✓	✓			✓
	Analysis of monitored data	✓	✓			✓
	Act on analysed data	✓				

- **End-to-end encryption is supported**: Communication uses Zigbee which supports end-to-end encryption. However, we did not find any claims from the vendor about end-to-end encryption. We also did not experimentally verify this claim. Thus, this claim is partially trusted (50%) but has the plausibility of 100% if verified experimentally or claimed by the vendor.
- **Does not use custom encryption algorithms**: This sub-claim has 100% belief because the communication uses the Zigbee protocol with a standard AES 128-bit encryption.

2) Access Control: In our case, a C&C command is triggered based on a predefined threshold setting. The C&C command is sent from the IoT hub to the devices in the home network. We consider the following sub-claims to fulfil P4:

- **Weak and default credentials are not allowed**: The hub and the devices are authenticated using pre-shared unique keys allowing only authorized nodes access to C&C data. The C&C data has restrictions to be accessed and triggered only by the gateway. Thus, we consider access control as adequate and assign the belief of 100%.
- **Enable Multi-factor Authentication**: This claim is not relevant because the control signals are sent autonomously and thus user authentication is not involved.

3) Monitoring and analysis: The claim for this criterion can be supported by the following two sub-claims:

- **C&C data is adequately monitored**: The SHEMS in our context supports basic monitoring. The log information such as devices status and control signals are collected. Thus, the assigned belief is 98% because we have not done testing in the lab of the logging system for bugs.
- **C&C data is adequately analyzed**: The gateway performs regular availability check on its devices and notifies about the disconnection of device(s). Though it is possible to manually analyze the monitored data more thoroughly from the log, such more extensive security analysis on collected data is not performed. Thus, the sub-claim has a lower belief set to 80%.

Table IV summarizes the beliefs, plausibilities and weights (w) assigned to the parameters for protection level evaluation of the selected criteria.

B. Aggregation using the weighted mean approach

Since there was no disbelief, and we calculate

$$Plausibility = 1 - Disbelief \tag{6}$$

then the plausibility in all cases was 100%.

Using the weighted mean approach (Equation 2), we calculated the aggregated belief for Data Encryption criterion as 89%, Access Control as 100% and Monitoring & Analysis as 89%. Further aggregation gave us 93% belief on the claim of protection level P4. Thus, we assign the overall confidence to the class A evaluation as 93% belief and 7% uncertainty, i.e. <A, 93, 7>.

C. Aggregation using MM approach

Using this approach, the aggregated belief for Data encryption, Access Control and Monitoring & Analysis criteria obtained were 78%, 100% and 86%. Similarly, the aggregated belief for P4 was 84%. Since plausibility was considered 100% all the time, it does not change after aggregation. The resulting class obtained was class A with 84% belief and 16% uncertainty i.e., <A, 84, 16>. Table V summarizes the results from the weighted mean and MM approach.

V. ANALYSIS AND DISCUSSIONS

We compared two approaches to aggregate beliefs. The weighted mean approach is not sensitive to low values. For instance, among the data encryption criteria from Table IV, there is one security functionality with weight 80 and belief 50. However, the aggregated belief is calculated as 89% in Table V. This value is not very realistic, because in security if one of the claims has low belief, it may have a high effect in the overall security (i.e., the "weakest-link" principle). Hence, the lower values should be well reflected in the aggregation of beliefs in security. When applying the MM approach to aggregate the belief for the same criterion (Data Encryption), we obtain the aggregated value of 78%, which is somewhat more realistic than 89%; suggesting the MM approach to aggregation of beliefs as preferable.

Because there is no disbelief in our case, the plausibility is 100%. Thus, the overall evaluation using the MM approach produced a belief of 84% and an uncertainty measure of 16%. The uncertainty can be reduced by providing missing evidence, e.g., the belief in the existence of end-to-end encryption can be increased by performing experimental validation.

TABLE IV: Belief, Plausibility and Weights on security claims.

Protection Criteria	**Security Functionality**	**<Bel, Pl, w>**
Data Encryption (w=100)	C&C data is encrypted between IoT hub and devices	<100, 100, 100>
	Data encryption uses a strong encryption algorithm	<100, 100, 95>
	End-to-end encryption is supported	<50, 100, 80>
	Does not use custom encryption algorithms	<100, 100, 95>
Access Control (w=95)	Weak and default credentials are not allowed	<100, 100, 100>
Monitoring and Analysis (w=80)	Monitor system components	<98, 100, 100>
	Analysis of monitored data	<80, 100, 95>

TABLE V: Comparison of weighted mean and Multi-Metrics approach for belief aggregation.

Weighted Mean Approach		**Protection Criteria**	**Multi-Metrics Approach**	
Aggregated to Protection Level	**Aggregated to Criterion Level**		**Aggregated to Criterion Level**	**Aggregated to Protection Level**
Bel = 93% Pl = 100% U = 7%	Bel = 89% Pl = 100% U = 11%	Data Encryption (w=100)	Bel = 78% Pl = 100% U = 22%%	Bel = 84% Pl = 100% U = 16%
	Bel = 100% Pl = 100% U = 0%	Access Control (w=95)	Bel = 100% Pl = 100% U = 0%	
	Bel = 89% Pl = 100% U = 11%	Monitoring & Analysis (w=80)	Bel = 86% Pl = 100% U = 14%	

Both uncertainty and disbelief increase unreliability. However, higher disbelief shows unreliability with certainty (obtained via evidence) indicating a weaker statement. As an example of disbelief, let us say that a claim is made for adequate data encryption for Wi-Fi communication using WEP standard. The disbelief in the claim is high and the uncertainty is low because, although WEP provides encryption, it is proven to be weak. Hence, to reduce the disbelief, the WEP must be upgraded to a more secured standard such as WPA2.

In the assessment, if the belief is too low, and the uncertainty is high, then the assessment requires more work or the experts may have less knowledge about the security built into the system. However, if the belief is low and the disbelief is high, it means the claims made in the assessment are not trustworthy. Therefore, appropriate measures should be taken to improve confidence. Similarly, an acceptable but not too high value of beliefs may say that the claims are trustworthy but not fully acceptable. In terms of security class, it may mean a different level of trust in the assessment. For instance, a claim of Class A with belief 60% and plausibility 95%, may mean a less trusted class A (which we could denote as A^{-}), while a belief of 95% may represent a highly trusted class A (e.g. A^{++}).

VI. CONCLUSION

We have shown how to extend security classifications with confidence parameters (i.e., belief and uncertainty) focusing on the methodology presented in [3]. We then exemplified the calculation of confidence parameters for a use case involving an edge command & control mechanism for SHEMS. We compared two types of methods for aggregating confidence measures, observing that weighted average methods are less suitable for security assurance than methods based on Root Mean Squared Weighted Data as the one used to aggregate "criticality" in the Multi-Metrics method of [14]. There are though different principles guiding when to consider minimum, maximum, or zero belief during aggregation.

Further work is needed for building these argumentation and aggregation methods into a tool, following works on tools like NOR-STA [10]. This work is more difficult to integrate with the security classification methodology that comes with predefined mechanisms and look-up tables.

REFERENCES

[1] Manish Shrestha, Christian Johansen, and Josef Noll. Building Confidence using Beliefs and Arguments in Security Class Evaluations for IoT (long version). Technical Report 493, Uni. Oslo, January 2020.

[2] Gianmarco Baldini, Antonio Skarmeta, Elizabeta Fourneret, Ricardo Neisse, Bruno Legeard, and Franck Le Gall. Security certification and labelling in internet of things. In *IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pages 627–632. IEEE, 2016.

[3] Manish Shrestha, Christian Johansen, Josef Noll, and Davide Roverso. A methodology for security classification applied to smart grid infrastructures. *International Journal of Critical Infrastructure Protection*, 28:100342, 2020.

[4] Manish Shrestha, Christian Johansen, and Josef Noll. Criteria for security classification of smart home energy management systems. In *Int. Conf. Smart Information & Comm. Technologies*. Springer, 2019.

[5] Jean Gordon and Edward H Shortliffe. The dempster-shafer theory of evidence. *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*, 3:832–838, 1984.

[6] Trudy Govier. *A practical study of argument*. Cengage Learning, 2013.

[7] Peter Bishop, Robin Bloomfield, and Sofia Guerra. The future of goal-based assurance cases. In *Assurance Cases*, pages 390–395, 2004.

[8] Tim Kelly. Reviewing assurance arguments-a step-by-step approach. In *Workshop on assurance cases for security-the metrics challenge, dependable systems and networks (DSN)*, 2007.

[9] John Goodenough, Howard Lipson, and Chuck Weinstock. Arguing security-creating security assurance cases. *rapport en ligne (initiative build security-in du US CERT), Université Carnegie Mellon*, 2007.

[10] Lukasz Cyra and Janusz Gorski. Support for argument structures review and assessment. *Reliability Eng. & System Safety*, 96(1):26–37, 2011.

[11] John Spriggs. *GSN – The Goal Structuring Notation: A Structured Approach to Presenting Arguments*. Springer, 2012.

[12] Stephen E. Toulmin. *The uses of argument*. Cambridge Uni.Press, 2003.

[13] Rui Wang, Jérémie Guiochet, Gilles Motet, and Walter Schön. Safety case confidence propagation based on Dempster–Shafer theory. *International Journal of Approximate Reasoning*, 107:46–64, 2019.

[14] Josef Noll, Inaki Garitano, Seraj Fayyad, and Habtamu Abie. Measurable security, privacy and dependability in smart grids. *Journal of Cyber Security and Mobility*, 3(4):371–398, 2014.