Data Protection Fortification: An Agile Approach for Threat Analysis of IoT Data

Sigrid Marita Kvamme*
Espen Gudmundsen*
Norwegian University of Science and Technology
Trondheim, Norway
sigridkvamme@gmail.com
oboygutt93@gmail.com

Tosin Daniel Oyetoyan
Western Norway University of Applied Sciences
Bergen, Norway
tosin.daniel.oyetoyan@hvl.no

Daniela Soares Cruzes
Norwegian University of Science and Technology
Trondheim, Norway
daniela.s.cruzes@ntnu.no

Abstract

Data from Internet of Things (IoT) devices has become a critical asset for decision-making. However, IoT devices have security challenges due to their low-resource constraints, heterogeneity, and deployment in hostile environments. Systems consuming IoT data must thus be designed with security measures to detect and prevent data tampering attacks.

We develop a data-centric threat modeling method named Data Protection Fortification (DPF) that practitioners can use during planning to assess and mitigate the security risk of using IoT data sources. We use design science to develop and validate DPF on 5 development teams from 3 organizations.

Results show that DPF can be used to identify and improve security practices of data sources. Practitioners have a positive attitude towards using DPF and because it is easily understood, it has the potential to become a communication tool for security between developers and stakeholders.

Keywords: IoT, Data-centric Threat analysis, Data protection fortification, Security, Threat analysis

ACM Reference Format:
Sigrid Marita Kvamme, Espen Gudmundsen, Tosin Daniel Oyetoyan, and Daniela Soares Cruzes. 2022. Data Protection Fortification: An Agile Approach for Threat Analysis of IoT Data. In *IoT '22: The 12th International Conference on IoT, Nov 07–10, 2022, Delft, NL.* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Data from IoT devices is increasingly prevalent and has become a critical asset for decision-making in different domains such as smart grids, smart city and smart ocean [9, 10]. However, ensuring that the data has not been tampered with by an adversary that aims to influence decisions is difficult, as IoT comes with several challenges. IoT devices have limited resources and are restricted to light-weight encryption mechanisms and authentication algorithms. Furthermore, they are typically deployed in unattended environments where an attacker can physically take over the device [3], and potentially make it send maliciously altered data [3, 10] or carry out injection attacks through the data. In addition, certain IoT domains such as the marine may rely on underwater acoustic communication that has bandwidth limitations and thus limit the volume of data to transmit and the security measures to employ[9]. Therefore, systems consuming such data must be designed with security measures to detect and handle such threats. With an increased focus and reliance on big data from smart devices, we argue that threat analysis methods specifically tailored for data is needed.

We develop a data-centric threat modeling method, named DPF, that focuses on data using inspiration from Protection Poker [12], and that is applicable during the design phase of secure software development life cycle (SSDLC). The goal of DPF is to help identify security measures on IoT data and determine what IoT data to be monitored for threats. We evaluate DPF and provide answers to two research questions:

- **RQ1** What are the practices reported by companies for securely handling data coming from devices or other sources?
- **RQ2** To what extent is DPF useful and accepted among practitioners?

The paper is structured as follows: Section 2 describes the methodology. In Section 3, we describe the DPF method. Section 4 presents the results of evaluating DPF with some discussion. The paper concludes with Section 6.

2 Methodology

We have conducted a study with three different organizations namely; Org1, Org2, and Org3. Table 1 describes the demography of the teams participating in this study.

2.1 Approach for Research

We have employed a framework described in [4] for carrying out design science research (DSR) [8] projects.

2.1.1 DPF Design and Data Collection. The teams use agile development methodology which motivated us to adapt Protection Poker for threat analysis of data sources used by their systems. We facilitated 5 DPF sessions with 5 different teams in focus groups, with each session lasting 1,5-2 hours. We did the sessions in three ways to adapt to the teams: (1) fully remotely through Microsoft Teams, (2) co-located using

*Both authors contributed equally to this research.

Table 1. Demography of participants. *n* = no. of participants.

Team	n	Description
Org1	13	An international energy company with over 21 000 employees in almost 30 countries
Org1-Team1	5	Offers a data platform that integrates various data sources related to marketing and midstream processing.
Org1-Team2	3	Receives data from Org1-Team1 and maintains a platform for their users to process data or create machine-learning models for gas trade recommendations.
Org1-Team3	5	Visualizes 3D models of Org1-governed installations, such as oil platforms and ships, and makes them available to different environments and users.
Org2	2	Market leader in micro-mobility services in one of the EU countries, operating a fleet of several thousand electric scooters. Org2 processes data generated from users and IoT devices.
Org3	9	A committee with the technical responsibility for a student organization. The data handled by Org3 contains personally identifiable information (PII) of students. Hence, data handling must be secure and follow GDPR regulations.

digital tools, and (3) co-located (with one remote member) using physical tools. We facilitated 3, 1, and 1 session of these, respectively, and the method was improved iteratively based on DSR. The method is grounded in relevant security sources[2, 7, 12]. We involved two security experts in separate meetings to evaluate its potential, the validity of the questions and the security measures presented. Using the Technology Acceptance Model (TAM) [1], we evaluate the participants' perceived ease of use, perceived usefulness, and acceptance of the method using a scale of 1-5, where 1 corresponds to *Strongly disagree* and 5 corresponds to *Strongly agree*. The final DPF method is described in Section 3.

2.1.2 Data Analysis.

To answer RQ1, we performed thematic coding[11] and evolved categories from the qualitative data. The practices and characteristics found in each category were marked based on how they might affect the risk of data tampering. We based this on best practices (and examples of misuse) from [6], OWASP Proactive Controls Guide[1] and our own experience as developers and security researchers.

To answer RQ2, we analyzed the evaluations (TAM and DPF activities) from the DPF sessions. The qualitative data from the oral evaluations were summarized, and subsequently analyzed with inductive coding. The codes were grouped by the different parts of the method (discussion, prioritization, security measures), and the relevant TAM categories.

3 Data Protection Fortification

Figure 1 illustrates the three main processes included in the method and the artifacts generated by each.

3.1 Preparations for DPF

When preparing the DPF activity, there are a few relevant questions to consider. An example is: *"What data source should be in focus?"*.

When to Perform DPF: We propose to make DPF a routine when assessing the security impact of integrating a new or existing data source to a system. It should be repeated for a data source when changes in the architecture affect how the data triggers transactions or decisions in the application or when the data fields change.

The team first selects a data source and then prepare a list of the known data fields, including relevant data types.

Participants: DPF is designed for development teams and other stakeholders such as; the project manager, product owner, security experts, testers. We recommend to include someone with in-depth knowledge about the data source and its usage.

Materials for DPF: We have created a toolbox containing materials for DPF[2] including MS Word templates, security measure glossary, Miro (digital whiteboard), guides, security questions, cheat sheets, and Mentimeter for collecting votes.

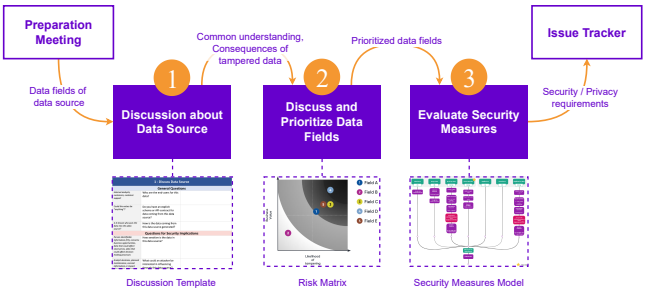

Figure 1. Overview of the 3 steps in DPF, with their input/output relationship, and artifacts generated at each step.

3.2 How to Perform DPF

Discuss Data Source: To start with, the participants discuss general information about the data source. Then, the participants discuss possible security implications for the data source. Question guide for this activity can be found in the DPF materials. An example is *What describes the data we receive from this data source? (e.g., gas prices, scooter info).* An example of security questions is *What could be the goal of an attacker that tampers with this data source? (e.g., influence analyst decisions or damage company reputation).*

Identify and Prioritize Data Fields: This step includes discussing how each data field influences the system, which can be as simple as mapping the field to its usage. For example, *temperature_celsius* influences production of energy.

Estimation is then performed to prioritize data fields. The estimation of *business value* is based on *value of the asset* while *likelihood of tampering* is based on *attack exposure.*

The participants should first calibrate all data fields before estimation to give a sense of which fields have the highest and lowest values for business value and the likelihood of tampering. To estimate the *business value* of a data field, some questions should be considered (see the DPF Materials). An example of such question is: *"How are functions, humans,*

[1] https://owasp.org/www-project-proactive-controls/

[2] tinyurl.com/2p99mxaw

or internal services dependent on this data field?". Also, to estimate the likelihood of tampered data within the data fields some questions should be considered to guide estimation. For example, *"Is the source of this field coming from human input, IoT devices, or is it system-generated?"*.

After, a risk matrix is created based on the values from the previous estimation activities using the risk formula 1.

$$\text{DPF Risk} = \text{Business Value} \times \text{Likelihood of Tampering} \quad (1)$$

Finally, we identify security measures that should be applied to the fields in the data source, using the method in Figure 2. It involves two primary artifacts, (A) a toolbox that contains security measures for commonly used data types, (B) a visualization of data fields in the data source with "lanes" for adding security measures. Participants actively use the toolbox to find relevant security measures for each data field in a data source. Security measures can be prioritized for the data fields with the highest risks.

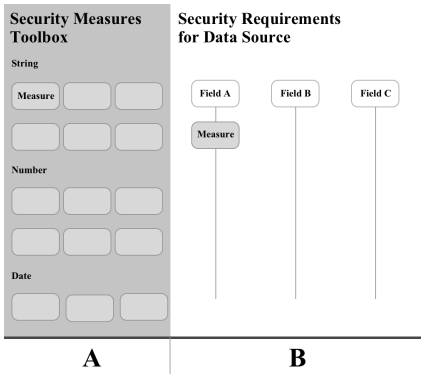

Figure 2. Method of evaluating security measures

Security Measures Toolbox We have curated a list of security measures for some commonly used data types such as *String* (textual values), *Number* (numerical values), and *Date* (date, with or without timestamps and timezone). We have also included security measures for the data source as a whole (see the DPF Materials).

4 Results and Discussion

4.1 RQ1: What Are the Practices Reported by Companies for Securely Handling Data Coming From Devices or Other Sources?

We present some of the results derived from performing the first step of DPF with the teams in Table 2. Subsequently, we provide evaluation of two practices (Data validity/Monitoring and data usage) and how they contribute to the risk of data tampering.

Data Validity and Monitoring: Three out of five teams reported implementing some measures for ensuring the integrity of the data they receive. Syntactic validations can reduce the risk of "low hanging fruit" tampering efforts, such as sending large amounts of text in an input field to cause a denial of service. Semantic validations can ensure the correct behavior and use of a system and may reduce the risk of subtle tampering attacks that syntactic validations can not easily identify. System designers and architects should define semantic validations related to how the system will be or is used. One type of monitoring implemented by one team to detect tampering is to survey QR-code scannings, where failed scans are sent for auditing, along with where and when the scan happened. The logs are visualized in a geographical heat map to allow customer service monitor if QR-code scans fail in a particular city or district. Physical tampering with the devices is also detected via a gyroscope that detect any movement when the device is not in use.

Table 2. Examples of reported practices. ✅ = Good, 🚩 = Should be followed up, 🚩 = Bad, ⛔ = Does not affect risk

Category	Practice	Evaluation
Data Access	Part of team can edit connection URL	✅
Data Access	Editing connection URL in cloud config is audited	✅
Data Access	Connection URL is placed in database	🚩
Data Access	Connection URL is placed directly in code	🚩
Data Validity	Prepared Statements (SQL) on retrieval	✅
Data Validity	No sanitization on storage	🚩
Data Validity	No validation	🚩
Data Velocity	Receives data continuously	🚩
Data Origin	System-generated data	✅
Data Origin	Human input	🚩
Data Origin	IoT-device data	🚩
Data Processing	Front-end framework with injection protection	✅
Data Processing	Exported to CSV	🚩
Data Schema	Explicit schema with API contract	✅
Data Schema	Notified if schema changes happen externally	✅
Data Schema	Text data type mostly used (even on numbers, dates)	🚩
Data Usage	Presented directly (raw format)	🚩
Data Usage	Used in machine learning	⛔
Data Usage	Used in device decisions	⛔
Fault Mitigations	Logging data that goes into ML model	✅
Fault Mitigations	External users report problems	🚩
Fault Mitigations	Operations monitored manually	🚩
Fault Mitigations	No routines to detect faults or handle it	🚩

Data Usage: One team did not process the data due to sensitivity concerns and presented it directly to the users. One data source was used in machine learning in one application and aggregated and presented to a user through a dashboard in another. Another data source was used by IoT devices to make decisions about how the device should function. Presenting raw data without using any safe viewing framework or prior validation could pose security risk.

4.2 RQ2: To what extent is DPF useful and accepted among practitioners?

4.2.1 TAM Evaluation of DPF.

Figure 3 shows a radar chart of the TAM results, where the units indicate how strongly each team, on average, agreed to each statement. *Use of Mentimeter* and *Useful discussions* received the highest scores on average (4.9 and 4.8), while *Fits team* and *Increased competence* received the lowest average scores (3.6 and 3.9). The average score for all the questions was 4.5.

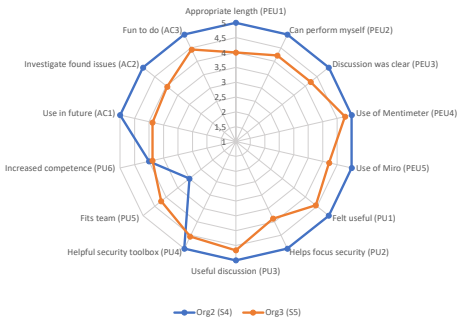

Figure 3. TAM evaluation results from Org2 and Org3.

4.2.2 Evaluation of the Activities in DPF.

We asked the participants in each session to rank the three activities in DPF they felt was the most useful. Ten participants answered from both companies. Figure 4. shows that *Evaluate Security Measures* and *Prioritize Data Fields* were perceived as almost equally useful, with only one vote in favor of the former. *Data Source Discussion* received significantly lower votes, where only one participant felt it was the most useful step.

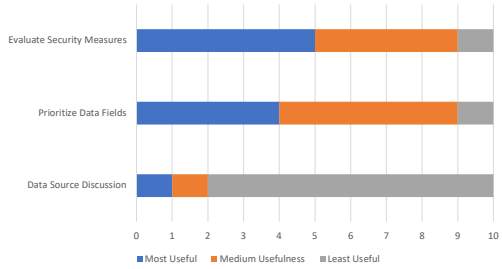

Figure 4. Participants evaluation of DPF.

4.3 Implications

Developers had a positive attitude towards focusing the session on fortifying the security of data fields in a data source, as they were more recognizable to them. More research on methods for data-focused threat analysis is encouraged.

Further work is needed to find an appropriate way to make the security measure diagram more maintainable and closer to the implementation code. For this, a structured format such as JSON or Markdown could be used, which could be checked into the version control software.

5 Threats to Validity and Limitations

The research paradigm chosen for this research is interpretivism [5]. There may thus be multiple ways to interpret our data and multiple conclusions to draw to answer our research questions. The scope of DPF is limited to threats related to data tampering, from the point of view of a data consumer. In future work, it can consider other threats than data tampering. Another limitation is the relatively low number of data fields that are viable to cover in one DPF session.

6 Conclusion

DPF as a data-centric threat analysis method can aid development teams to evaluate security risk of an external data source and provide security measures. Addressing security from a low-granularity data-oriented perspective showed promise of acceptance among practitioners. Developers felt familiar with looking at securing data fields and discussing their implications. Contrary to many other threat modeling methods, DPF shows the potential not to require a security expert to facilitate the meeting. Applying DPF in the design phase equips the development team with a structured method to iron out possible assumptions about the data. Unless addressed, these assumptions may introduce security challenges when implementing functions using the data.

Acknowledgments

Tosin Daniel Oyetoyan was supported by the SFI Smart Ocean Project 309612/F40.

References

[1] Fred D Davis. 1985. *A technology acceptance model for empirically testing new end-user information systems: Theory and results.* Ph. D. Dissertation. Massachusetts Institute of Technology.

[2] Shawn Hernan, Scott Lambert, Tomasz Ostwald, and Adam Shostack. 2019. *Uncover Security Design Flaws Using The STRIDE Approach.* https://docs.microsoft.com/en-us/archive/msdn-magazine/2006/november/uncover-security-design-flaws-using-the-stride-approach

[3] Da-Wen Huang, Wanping Liu, and Jichao Bi. 2021. Data tampering attacks diagnosis in dynamic wireless sensor networks. *Computer Communications* 172 (2021), 84–92.

[4] Paul Johannesson and Erik Perjons. 2014. *An introduction to design science.* Vol. 10. Springer.

[5] Briony J Oates. 2006. *Researching Information Systems and Computing.* Sage.

[6] OWASP. 2021. *Cheat Sheet Series.* https://cheatsheetseries.owasp.org/

[7] OWASP. 2021. *Threat Modelling Process.* https://owasp.org/www-community/Threat_Modeling_Process

[8] Ken Peffers, Tuure Tuunanen, Marcus A Rothenberger, and Samir Chatterjee. 2007. A design science research methodology for information systems research. *Journal of management information systems* 24, 3 (2007), 45–77.

[9] Tie Qiu, Zhao Zhao, Tong Zhang, Chen Chen, and CL Philip Chen. 2019. Underwater Internet of Things in smart ocean: System architecture and open issues. *IEEE Transactions on Industrial Informatics* 16, 7 (2019), 4297–4307.

[10] Haftu Tasew Reda, Adnan Anwar, and Abdun Mahmood. 2022. Comprehensive survey and taxonomies of false data injection attacks in smart grids: attack models, targets, and impacts. *Renewable and Sustainable Energy Reviews* 163 (2022), 112423.

[11] Colin Robson and Kieran McCartan. 2016. *Real world research: a resource for users of social research methods in applied settings.* Wiley.

[12] Laurie Williams, Andrew Meneely, and Grant Shipley. 2010. Protection poker: The new software security "game". *IEEE Security & Privacy* 8, 3 (2010), 14–20.