Trust Patterns In Modern Web-API-based Service Architectures - More Than Technical Security Aspects

Sandro Hartenstein
Berlin School Of Economics And Law
Berlin, Germany
email: sandro.hartenstein@hwr-berlin.de

Steven Schmidt
DB Station&Service AG / Berlin School Of Economics And Law
Berlin, Germany
email: s_schmidts19@stud.hwr-berlin.de

Andreas Schmietendorf
Berlin School Of Economics And Law
Berlin, Germany
email: andreas.schmietendorf@hwr-berlin.de

***Abstract*— This idea paper describes the current perception of Security Patterns and the authors view on the need, to broaden this technical view to a more wholesome approach – Trust Patterns, *integrating* security features. Explaining the need for this approach, it is shown, how this would influence the user's perception of security features through trustworthiness aims towards a consumed service.**

Keywords-Trustworthiness; Digitalization; Information Systems; Security Management; Society.

I. Introduction

This idea paper aims on emphasizing the importance of trust and trustworthiness of systems in contrast of solely secure designs in a functional way. Security Patterns should be taken into consideration when utilizing Trust Patterns, but a broader view beyond technological aspects into socio-technical or even sociological sides of security and correlating trustworthiness enables a richer and sustained effect and impact towards the user.

Hill and O´Conner define trust in their journal article *A Cognitive Theory of Trust* as follows:

"Trust by definition entails a willingness by the [trustor] to make herself vulnerable to the possibility that another will act to her detriment" [1, p. 28]

A large part of the rapid digitization of services is enabled by the use of WebAPIs. By orchestrating partial services into a full application via apis, it is possible to reduce the effort required compared to a full implementation of all aspects. With this setting, the WebAPIs must be trustworthy in order to be successful. The digitization depends on the well-being of the users. So trustworthy apis are needed, especially due to the rising complexity and in transparency of current and emerging digital services. Trust towards a WebAPIs generates a higher likeliness of using the WebAPI regularly, repeatedly and by recommendation, which are all factors aside from classical security aspects. To relate to the previous quote: Improving trustworthiness not only by security measures but a broader and whole view, increases the chances of consumption and usage.

This paper has the following structure. In the second section, the initial definitions and views are given. In the third section, the required preliminary work is explained. In the fourth section the concept is presented. In the fifth section, the main facts are briefly explained and a planned research project on this topic is presented.

II. Terms and viewpoints

A pattern is an idea that has proven itself in one practice and is likely to be useful to others. There are security design patterns that address typical security challenges and there are trust patterns that address typical trust antecedents.

A pattern typically addresses the process, product, and/or resources. For example, there are security patterns for encrypted transport of data. The communication over this encrypted connection to the user is not part of the pattern. However, the user needs this information to build up trust and to recognize the value subjectively. Therefore, from our point of view, the trust pattern for encrypted transport of data consists of the technical security part and the communicative promotion part in consequence.

Further security pattern deals with the authorization of a webapi access, like the token-based OAuth approach. Another aspect is related to a federated identity management like the application of SAML (Security Assertion Markup Language) or the implementation of a single sign on approach with OpenID and Keycloak.

A trust pattern can exist without a technical part. For example, the reputation of the service provider strongly influences the trust towards his services [2].

The difficulty of creating trust patterns is, that the impact of trust solutions is difficult to prove. The effects that create trust are far more complex than, for example, security, and thus harder to measure. Plus, trust building measures have not always been implemented explicitly, if even. Due to this, a record of past implementations and their possible successful impact will be hard to determine.

Patterns typically addresses the process, product, and/or resources. Trust patterns also address all dimensions and should cover trust in a holistic way.

III. Related Work

Patterns characterised by [3, p. 3] as follows: A pattern is both a spatial configuration of elements that solve a particular problem and a set of associated instructions to create that configuration of elements as effectively as possible. Patterns represent proven and optimal solutions to given problems. This assumes that these solutions and concepts have been successfully applied again and again in the past.

In order for patterns to become successful or resilient, they must be evaluated [4, p. 4].

For this purpose, the Patterns are evaluates after each step in the lifecycle, defined by [5]. The lifecycle begins with the theory and the specific domain knowledge from which a pattern is developed. This is then deployed and applied. The experiences from the application in use are used in the development [5].

In the development phase, the evaluation is carried out with expert review. In the deployment phase, evaluation takes place with a workshop and peer review. In the operational phase, experiments and surveys are used to check the requirements for patterns [4, p. 5].

Hoffmann's research team published twenty Trust Patterns in 2012.These patterns are templates for defined requirements. For example, a trust pattern, named data usage, is the provision of information on how data is used by the system for the recommendation. Another trust pattern is, for example, the self-explanatory button icon, which states that a button correctly describes the further behaviour of the system. The trust pattern, named setting options, is the provision of personal settings to customise the system [6, pp. 8-10].

These patterns are based on the influencing factors, called Antecedents of Trust, of Söllner et. al, Lee and See and Muir. In relation to the Trust Patterns examples, Understanding, Predictability and Personalisation for the user are the respective arguments [7] [8] [9].

23 principles and 47 patterns for trusted user interfaces has been compiled and prepared in 2018 [10]. The interactive online repository contains not only the content of the patterns, but also meta-information about their origin and links to other patterns and principles [11]. It is also available via webapi, so that it can be easily integrated into development environments. A good example is *Warn When Unsafe*. This pattern addresses informing the user when the configuration of the system is unsafe. It provides for the user to be informed periodically. The frequency of the warnings is very important so that the user notices it but does not get used to it. This is implemented by monitoring the configuration and a safe reference value. Linked patterns are *Attactive Options, Immediate Notifications, Conveying Threats & Consequences, General Notifications About Security, Immediate Options* and *Separating Content*. This pattern originated in Garfinkel's PhD thesis [12] [13].

In summary, it can be said that trust patterns are already being developed and applied in some areas, such as marketing and user interface, due to economic interests. Also, the technical security aspects are also mostly already researched and published. From our point of view, a holistic approach to WebAPIs is missing, which is necessary for the establishment of trustworthy WebAPIs.

IV. Conceptual considerations

Our conceptual reasoning is that a holistic, multidimensional view of the trustworthiness of WebAPIs can add great value to digitization. Patterns provide a good way to address non-functional and functional requirements for developing, marketing, and communicating WebAPIs.

Following the trust aspects of software, shown in Figure 1, the product related trust patterns should address the WebAPI relevant ones. Applied security mechanisms such as OAuth 2 address the attributes confidentiality and non-repudiation. The composability is characterized by patterns that reveal the degree of coupling and possibly also the dependencies at the interface. Other features relate to the data processing of WebApis' downstream algorithms, for example, a verification pattern may require the transmission of hash values, thus promoting data integrity.

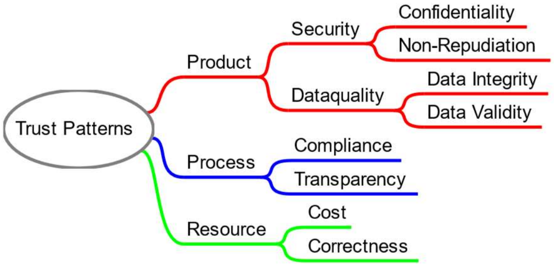

Figure 1 Classification of with the trustworthiness attributes from [14, p. 547].

Process-related trust patterns are intended to the way WebAPIs and downstream services are created. For example, patterns of information sharing about the development process, as well as vulnerability management, are purposeful. Requirements for process certification and specification can also have a confidence-building effect and should be offered as patterns. Another example is patterns for disclosing roadmaps of WebAPIs so that developers can prepare for possible API deactivation or serious changes.

Resource related Trust Pattern do not directly address the WebAPIs, but rather the environment. For example, the corporate brand is an important indicator of trustworthiness and thus of trust. The usage behavior, the number of users, as well as the user types of a WebAPIs is also relevant for trust and should be enhanced with appropriate patterns. The scalability of a WebAPI will also contribute to its distribution and usage, thus promoting trust, should be addressed with patterns.

In a next step, the idea of building a catalog of trust patterns is pursued by carrying out an empirical study. This study is divided into several areas. On the one hand, the most important WebAPIs available are to be examined and, on the other hand, a survey of consumers (typically software developers) of WebAPIs is to provide information about the priorities of the choice of use.

V. Conclusion and Future Work

Testing trust in services is an important part of creating and establishing trust patterns. For this reason, trust-building measures should always be linked to an evaluation.

Müller and his research colleagues conducted a study on the impact of decentralized blockchain technology on trust in collaboration. This technological view confirms the connection between technology, understanding and trust. In this regard, further trust-oriented technologies should be investigated [15]. Assessments such as these motivate the action, to perform own examinations of the role of trust in various fields, which have typically only been conducted under security viewpoints in the past. To form a structured approach towards such examinations and findings, the Berlin School of Economics and Law founded a research project determined to an "*empirical evaluation of a model of trustworthiness*" (*orig.: Empirische Untersuchungen zur Modellierung von Vertrauenswürdigkeit*) – EUMoVe [16].

References

[1] C. A. Hill and E. A. O'Hara O'Connor, "A Cognitive Theory of Trust," *SSRN Journal*, 2005, doi: 10.2139/ssrn.869423.

[2] S. Schmidt, "Creating a trustworthy public WLAN - approach and partial results [orig.: Schaffung eines vertrauenswürdigen, öffentlichen WLANs - Herangehensweise und Teilergebnisse]," in *Berliner Schriften zu modernen Integrationsarchitekturen*, vol. 24, *ESAPI 2020: 4. Workshop Evaluation of Service-APIs*, A. Schmietendorf and K. Nadobny, Eds., 1st ed., Düren: Shaker, 2020, pp. 35–48.

[3] M. Schumacher, *Security patterns: Integrating security and systems engineering*. Chichester, England, Hoboken, NJ: John Wiley & Sons, 2006, ISBN: 978-0-470-85884-4. [Online]. Available: http://site.ebrary.com/lib/alltitles/docDetail.action?docID=10300660.

[4] A. Hoffmann, H. Hoffmann, and M. Söllner, "Fostering Initial Trust in Applications: Developing and Evaluating Requirement Patterns for Application Websites," *21st European Conference on Information Systems (ECIS), Utrecht, The Netherlands*, vol. 2013. [Online]. Available: https://www.alexandria.unisg.ch/228935/1/Hoffmann%20et%20al.%202013.pdf.

[5] S. Petter, D. Khazanchi, and J. D. Murphy, "A design science based evaluation framework for patterns," *SIGMIS Database*, vol. 41, no. 3, pp. 9–26, 2010, doi: 10.1145/1851175.1851177.

[6] A. Hoffmann, H. Hoffmann, and M. Söllner, "TWENTY SOFTWARE REQUIREMENTPATTERNS TO SPECIFY RECOMMENDERSYSTEMS THAT USERS WILL TRUST," *ECIS 2012 Proceedings.Paper 1*, 2012.

[7] M. Söllner and J. M. Leimeister, *15 years of measurement model misspecification in trust research? A theory based approach to solve this problem.* [Online]. Available: http://pubs.wi-kassel.de/wp-content/uploads/2013/03/JML_189.pdf [retrieved: 01, 2020].

[8] J. D. Lee and K. A. See, "Trust in automation: designing for appropriate reliance," *Human factors*, vol. 46, no. 1, pp. 50–80, 2004, doi: 10.1518/hfes.46.1.50_30392.

[9] B. M. MUIR, "Trust in automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems," *Ergonomics*, vol. 37, no. 11, pp. 1905–1922, 1994, doi: 10.1080/00140139408964957.

[10] L. Lo lacono, M. Smith, E. von Zezschwitz, P. L. Gorski, and P. Nehren, "Consolidating Principles and Patterns for Human-centred Usable Security Research and Development," in *Proceedings 3rd European Workshop on Usable Security*, London, England, Apr. 2018.

[11] L. Lo lacono, *USecured Tools.* [Online]. Available: https://das.th-koeln.de/usecured [retrieved: 01, 2021].

[12] S. L. Garfinkel, *Design Principles and Patterns for Computer Systems that are Simultaneously Secure and Usable*, 2005, ISBN: . Accessed: Feb. 1 2021. [Online]. Available: https://simson.net/thesis/thesis.pdf.

[13] L. Lo lacono, *Warn When Unsafe Pattern.* [Online]. Available: https://das.h-brs.de/usecured/patterns/warn-when-unsafe [retrieved: 02, 2021].

[14] N. Gol Mohammadi *et al.,* "An Analysis of Software Quality Attributes and Their Contribution to Trustworthiness," *Proceedings of the 3rd International Conference on Cloud Computing and Services, Science*, pp. 542–552, 2013.

[15] M. Müller, N. Ostern, and M. Rosemann, "Silver Bullet for All Trust Issues? Blockchain-Based Trust Patterns for Collaborative Business Processes," in *Lecture Notes in Business Information Processing, BUSINESS PROCESS MANAGEMENT: Blockchain and robotic process automation*, A. Asatiani et al., Eds., [S.l.]: SPRINGER, 2020, pp. 3–18.

[16] S. Hartenstein, S. Schmidt, and A. Schmietendorf, "Towards an Empirical Analysis of Trustworthiness Attributes in the Context of Digitalization," in *ICDS 2020 : The Fourteenth International Conference on Digital Society*, pp. 112–116.