Trust Awareness for Redecentralized Web Applications (Position Paper)

Valentin Siegert[1], Martin Gaedke[1]

[1]Technische Universität Chemnitz, Chemnitz, Germany

Abstract

Redecentralizing the web improves user privacy and data control, but also brings the challenge of acquiring trusted data across distributed data storage. In this paper we propose how redecentralized web applications can autonomously and automatically make trust-aware decisions about acquired data from decentralized stores, and we identify requirements and open challenges.

Keywords

Trust, Decentralized Web, Web Redecentralization

1. Introduction

Today, centralized web platforms store data in their controlled infrastructure, resulting in data silos and walled gardens [1, 2]. As this leads to significant privacy concerns [1, 3], initiatives such as Solid [1], the distributed social app Mastodon[1] or the EU's Next Generation Internet[2] are working to redecentralize the web. Achieving this redecentralization requires universal open protocols and application interfaces that allow users to be independent of centralized instances [1, 3, 4, 5]. Solid, for example, enables users to store and manage personal data in decentralized pods that contain linked data.

An imminent challenge for the redecentralized web is trusted data acquisition [2, 3, 5]. Since web application operators are interested in their users or customers perceiving the application as trustworthy, the web application must work with trusted data. In the current web, trustworthiness is determined by third parties based on predetermined artifacts or human-issued permissions created and controlled by centralized instances [5]. A redecentralized web application, on the other hand, works with data that is not hosted by it, but is available on any decentralized knowledge graph or solid pod. Since these are hosted by third parties, the web application has no control over the trustworthiness of the data, but must ensure it. Therefore, the proposed privacy and data freedom enhancements [1] may be limited if the data

TrusDeKW@ESWC'23: Trusting Decentralised Knowledge Graphs and Web Data Workshop at Extended Semantic Web Conference 2023, May 28 – June 1, 2023, Hersonissos, Greece

valentin.siegert@informatik.tu-chemnitz.de (V. Siegert); martin.gaedke@informatik.tu-chemnitz.de (M. Gaedke)
https://vsr.informatik.tu-chemnitz.de/people/siegert (V. Siegert);
https://vsr.informatik.tu-chemnitz.de/people/gaedke (M. Gaedke)
0000-0001-5763-8265 (V. Siegert); 0000-0002-6729-2912 (M. Gaedke)

CEUR Workshop Proceedings (CEUR-WS.org)

[1]https://joinmastodon.org/

[2]https://digital-strategy.ec.europa.eu/en/policies/next-generation-internet-initiative

acquired is malicious or harmful. Data is in this paper's context trustworthy if the using web application makes a positive trust-aware decision on whether to trust the data or not. This decision is taken based on an evaluation which computes factors with different input parameters and subsequent combination to one final trust value for the respective decision. Despite the influence of sociological and psychological research on trust, computing trust in the web focuses on algorithms, standards and empirical studies [6].

The redecentralized web can be metamodeled as an open and dynamic multi-agent system (MAS) [5] since its web applications are autonomous and structurally independent and the number of existing applications is large. Moreover, it classifies itself as an open and dynamic system since it is free of membership constraints and web applications can go on and offline at any time. The proposal of *hypermedia MAS* [7] envisages similarly the idea of allowing heterogeneous entities to interact uniformly through hypermedia. A web application (a web agent in MAS) can make autonomous trust decisions in many ways, as discussed in several research papers in the area of trust within MAS [5, 8] as well as within the web [6, 9].

In this position paper, we outline a proposal for making web applications trustworthy in an autonomous and automatic way, with the goal of acquiring web data from decentralized knowledge graphs, third-party Solid pods, or other decentralized stores of web data.

2. Redecentralized Web Applications' Trust Awareness

In a scenario where the web is redecentralized and data is stored in a decentralized manner in Knowledge Graphs (KGs), Solid Pods, and other distributed web data stores, a web application no longer stores data only on itself. Therefore, such a web application must be able to search the Semantic Web and dynamically acquire linked data on-the-fly. While data in the web might be still duplicated due to practical needs and responsive design, data has to be synced in both directions for privacy and data control purposes. A redecentralized web application will thus not always take data from a decentralized data store, but very often dynamically at runtime. Hence, the applications must constantly make trust-aware decisions on whether it should process received data or not. Those decisions are mandatory for only using trustworthy data to not decrease trust of users or customers in the web application by ignoring trust awareness. Yet, existing policies and norms should not be adapted, nor should other service qualities of web applications be negatively affected, e.g. its response time. Thus, our proposal does not contradict ideas about governance of autonomous agents on the web [10], but starts at the web application itself instead of defining governance on a global top-down view over (parts of) the web.

One example of such a scenario with decentralized data stores are redecentralized on-demand music streaming services. Some domain-related data such as recommendations for new discoveries, automatic playlists, or streaming functionality potentially remain at the individual applications. With the possibility of decentralized data storage, both users and music artists may have a high interest in storing their data in their own KG or solid pod. This allows data owned by users and artists to be reused across multiple services. However, because multiple applications read and write to these decentralized data stores, each application cannot ensure, even during the read process, that the data it reads is not harmful to itself. This can be due to data consistency issues caused by multiple applications working with the same data. Other

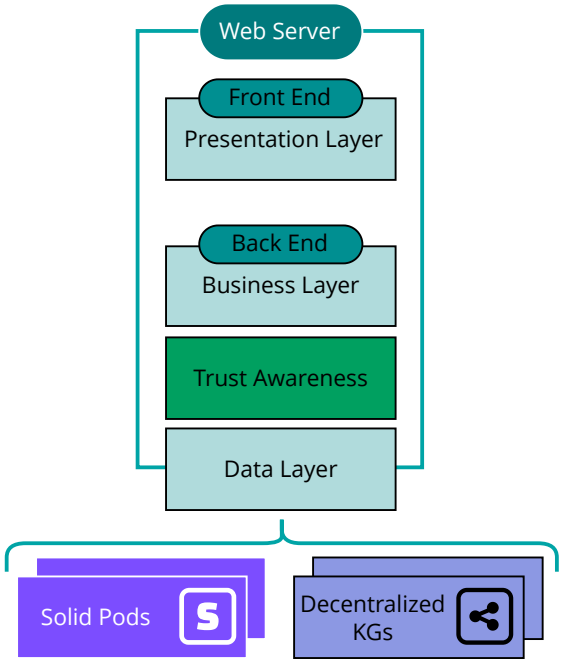

Figure 1: Trust Awareness in a Redecentralized Web Application. A 3-tier web application with redecentralized data layer and trust awareness layer between data und business layer.

reasons can be problems with data replication and fraudulent manipulation of data by competing services. A trusted decision about reading data from decentralized storage would reduce the risk of working with data that is not "good enough" for the purpose of one web application.

We propose that each web application should have its own trust awareness. Otherwise, redecentralization will soon be a centralized web again, but no longer dependent on centrally stored data, but on the provision of trust awareness or its testimonials. Trust testimonials in this context are artifacts that contain statements about the trustworthiness of other entities, be they hosts or data. We sketch in Figure 1 a common 3-tier webserver, which is redecentralized in its data layer and includes trust awareness. We indicate by positioning the Data Layer across the boundary of the web server, that the Data Layer is not necessarily a local or controlled layer of the web application anymore, but redecentralized. Additionally, the external sources of the data layer are Solid Pods, decentralized KGs or similar decentralized web data stores. Any local data storage might still be in place as parts of the Data layer is also still inside the web application. In the context of the newly added trust awareness, it needs to make decisions on which data is trustworthy enough to process and which one is not. Yu et al. [8] describes that the term trust awareness, can be divided in the two parts of *trust evaluation* and *trust-aware decision making*. Trust evaluation is the process to compute a numeric trust value or a cognitive ranking on given inputs as the interacted peer's identity, meta data of a message or even the message itself. The trust-aware decision making is then enabling the web application to decide about trustworthiness based on the results of the trust evaluation.

The idea we propose is consistent with Golbeck's view [6] in that trust on the semantic web

is critical for data and its exchange over decentralized data stores. However, Golbeck treats trust mainly as involving the user in the process of trust and its computation. This holds since user-to-machine or user-to-user trust relationships or manual trust conditions from users are considered in the aforementioned approaches of Golbeck. In this sense, Golbeck mentions for trust in web services mainly approaches for authorizing access to web service functionalities. In contrast, our proposal refers to a machine-to-machine trust relationship for the exchange of decentralized data. The most similar trust models in Golbeck's survey to our proposal are those for peer-to-peer (P2P) systems. However, these require at least adaptations towards the web's data heterogeneity. This includes the fact that most of these models assume an existing trust relationship path between all peers within the system. Our approach, on the other hand, assumes a high degree of individuality in the trust awareness of web applications, whether in terms of automation or autonomy.

3. Requirements and Open Challenges

To enable trust awareness within web applications, the following *requirements* apply: **Runtime:** To ensure adequate response time of a web application's back end, the runtime of the trust awareness needs to be as short as possible. Since the response times of web applications are within milliseconds to respond instantaneously[3], there is not much time to increase them with additional components like the proposed trust awareness. **Accuracy:** Any trust-aware decision should be as accurate as possible in order not to regret it later. Therefore, trust evaluations require high accuracy, which can be measured e.g. by comparing expected with calculated rankings of trust evaluations [11]. **Full Automation:** Trust awareness must be fully automated because the number of data sources on the web is already large and decentralization only increases the number of data providers [5]. Any human intervention would thus slow down the process to dynamically acquire data from decentralized data sources and overwhelm users with the amount of data sources available. **Autonomy:** Since the decentralization of the web should not be undermined by newly introduced central instances, web applications should handle trust autonomously. While considering trust testimonials can be valuable, outsourcing the complete trust evaluation would contradict our proposal. **Trust Attack Robustness:** As we introduce a component of trust awareness into a web application, the newly added component must be robust against known trust attacks [9, 12]. Similar to web application security attacks, trust awareness must not be compromised by misleading input.

The following *challenges* are the main to address to enable trust awareness for web applications: **Suitable Trust Model:** Although several trust models exist to date [2, 5, 8, 6, 9], it remains an open challenge to find the most suitable one or combinations of existent for the purpose of our presented scenario. **Initial Trust:** Initial evaluations for new applications can be as challenging as migrating existing collaborations between web applications. Trust awareness in web applications needs to establish initial values without full trust or full distrust at the outset. It remains an open challenge how to initialize trust evaluations by taking neither fixed default values nor random values that could influence following trust evaluations. **Web Data Heterogeneity:** The heterogeneity of web data poses additional challenges for trust

[3] https://www.nngroup.com/articles/response-times-3-important-limits/

evaluations, as information is only sometimes available, but information cannot always be collected in the same way, nor can it be collected everywhere.

References

[1] A. V. Sambra, E. Mansour, S. Hawke, M. Zereba, N. Greco, A. Ghanem, D. Zagidulin, A. Aboulnaga, T. Berners-Lee, Solid: A Platform for Decentralized Social Applications Based on Linked Data, Technical Report, MIT CSAIL & Qatar Computing Research Institute, 2016.

[2] V. Siegert, A. Kirchhoff, M. Gaedke, ConTED: Towards Content Trust for the Decentralized Web, in: Proceedings of WI-IAT 2022, 2022, pp. 604–611. doi:10.1109/WI-IAT55865.2022.00095.

[3] L. D. Ibáñez, E. Simperl, F. Gandon, H. Story, Redecentralizing the web with distributed ledgers, IEEE Intelligent Systems 32 (2017) 92–95. doi:10.1109/MIS.2017.18.

[4] M. Noura, V. Siegert, M. Gaedke, WAT: Autonomous Hypermedia-driven Web Agents for Web of Things Devices, in: Proceedings of the All the Agents Challenge co-located with the 20th International Semantic Web Conference, 2021, pp. 38–43. URL: https://ceur-ws.org/Vol-3111/short6.pdf.

[5] V. Siegert, M. Noura, M. Gaedke, aTLAS: A Testbed to Examine Trust for a Redecentralized Web, in: Proceedings of WI-IAT 2020, 2020, pp. 411–416. doi:10.1109/WIIAT50758.2020.00060.

[6] J. Golbeck, Trust on the world wide web: A survey, Foundations and Trends® in Web Science 1 (2008) 131–197. doi:10.1561/1800000006.

[7] A. Ciortea, S. Mayer, F. Gandon, O. Boissier, A. Ricci, A. Zimmermann, A decade in hindsight: The missing bridge between multi-agent systems and the world wide web, in: Proceedings of the International Conference on Autonomous Agents and Multiagent Systems, 2019. URL: http://www.alexandria.unisg.ch/256718/.

[8] H. Yu, Z. Shen, C. Leung, C. Miao, V. R. Lesser, A Survey of Multi-Agent Trust Management Systems, IEEE Access 1 (2013) 35–50. doi:10.1109/ACCESS.2013.2259892.

[9] Y. Ruan, A. Durresi, A survey of trust management systems for online social communities – trust modeling, trust inference and attacks, Knowledge-Based Systems 106 (2016) 150–163. doi:https://doi.org/10.1016/j.knosys.2016.05.042.

[10] T. Kampik, A. Mansour, O. Boissier, S. Kirrane, J. Padget, T. R. Payne, M. P. Singh, V. Tamma, A. Zimmermann, Governance of autonomous agents on the web: Challenges and opportunities, ACM Trans. Internet Technol. 22 (2022). doi:10.1145/3507910.

[11] D. Jelenc, R. Hermoso, J. Sabater-Mir, D. Trček, Decision making matters: A better way to evaluate trust models, Knowledge-Based Systems 52 (2013) 147–164. doi:10.1016/j.knosys.2013.07.016.

[12] Y. L. Sun, Z. Han, W. Yu, K. R. Liu, A trust evaluation framework in distributed networks: Vulnerability analysis and defense against attacks, in: Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications, 2006, pp. 1–13. doi:10.1109/INFOCOM.2006.154.