Customized Views on Profiles in WebID-Based Distributed Social Networks

Stefan Wild, Olexiy Chudnovskyy, Sebastian Heil, and Martin Gaedke

Chemnitz University of Technology, Germany
{firstname.lastname}@informatik.tu-chemnitz.de

Abstract. WebID as an extensible and distributed identification approach enables users to globally authenticate themselves, connect to each other and manage their identity data at a self-defined place. Identity data stored in WebID profile documents can be protected from unauthorized access using appropriate access control methods. While existing methods are primarily about securing resources, they lack providing adequate mechanisms for controlling access to specific data *within* profiles.

This paper presents our approach to create customized views on profiles in WebID-based distributed social networks. We introduce fine-grained personalized filters based on SPARQL templates and demonstrate their integration into an existing identity management platform.

Keywords: Social Web, Semantic Web, WebID, Access Control.

1 Introduction

Centralized social networks such as Facebook, Google+ or LinkedIn provide varied possibilities for personal information exchange, but try to bind users within their own domains [6]. Avoiding data silos and enabling users to remain in control of their data asks for a distributed social network (DSN). A DSN can be implemented on the basis of W3C's WebID specification [4]. WebID as a universal identification mechanism enables authentication through a client certificate that includes an URI, called WebID, referring to a resource containing the identity owner's data, called WebID profile. WebID profiles are extensible and machine-readable through RDF and domain specific vocabularies like FOAF. It is in the user's interest to consolidate personal data at one place and publish it in a uniform way to enable data reuse across different services and Web applications.

Unprotected WebID profiles, however, are potential information sources for known and wanted, but also for unknown and unwanted requesters. Authenticating via WebID requires a publicly accessible profile as it contains the profile owner's public keys, i.e., also data irrelevant to authentication per se could be retrieved [2]. Existing mechanisms only provide coarse access control and require outsourcing profile data to be protected as separate resources. There is a clear need for fine-grained and user-defined access control of WebID profile data.

We identified 3 requirements a solution has to fulfill: First, identity owners must be enabled to express fine-grained views on WebID profiles targeting different requesting agents. Second, view definitions must be portable to other systems

F. Daniel, P. Dolog, and Q. Li (Eds.): ICWE 2013, LNCS 7977, pp. 498–501, 2013.

without making major adjustments. Third, views on profiles must be standard compliant to ease maintenance, ensure traceability and reliable processing.

This paper addresses these requirements by the following contributions: First, we propose a flexible approach to customize views on WebID profiles using fine-grained filters. Second, we present an RDF-based filter language using SPARQL. Third, we demonstrate the integration of view customization facilities into an existing WebID identity provider and profile management platform.

The rest of this paper is organized as follows: We analyze related work in Section 2, present our solution in Section 3, and conclude the paper in Section 4.

2 Related Work

Web Access Control (WAC) enables access control to resources at the document level and supports assigning access rights to agents identified by WebIDs [2]. Access control lists specified by WAC are machine-readable through RDF and can be stored as self-contained resources independently from the resources they control access to. While WAC is well-suited for scenarios with many resources to be protected, it lacks possibilities to secure specific data within resources [1]. A fine-grained control requires outsourcing specific profile parts as separate resources and defining corresponding ACLs, which entails declining portability.

Similar to WAC, the Access Control Ontology (ACO) can only control access to resources [3]. ACO is more flexible than WAC, as it additionally supports defining roles and enables to directly map permissions to HTTP verbs.

The approach proposed in [5] enables manipulating profile data for specific requesting agents. Relevant profile data is addressed through URIs or RDF triple elements and logic defined by a custom vocabulary is interpreted to establish diverse views on profiles for specific requesting agents. Using a custom vocabulary limits expressiveness and portability. View definitions offer alternative information sources relative to existing WebID profile data. This requires further processing to merge or replace specific triples. Like ACO and WAC, treating particular profile data independently requires outsourcing as separate resources.

In contrast to the presented techniques, our approach enables filtering at the level of identity attributes while avoiding to distribute profile data.

3 Customized Views on WebID Profiles

To create customized views on WebID profiles, our solution automatically selects a filter specified for the requesting agent. If no filter specification is available, an identity fallback function retrieves the most appropriate filter. The entire WebID profile is converted via a graph-to-graph transformation into a filtered profile containing only data satisfying the visibility constraints defined for the requesting agent. For the transformation, we use SPARQL CONSTRUCT statements to apply a whitelisting to particular WebID profile data. The representation of the filtered profile is then sent back to the requesting party. Figure 1 illustrates this approach. Filters are created by WebID profile owners and stored in a machine-readable way within the profiles. As an RDF-based language using

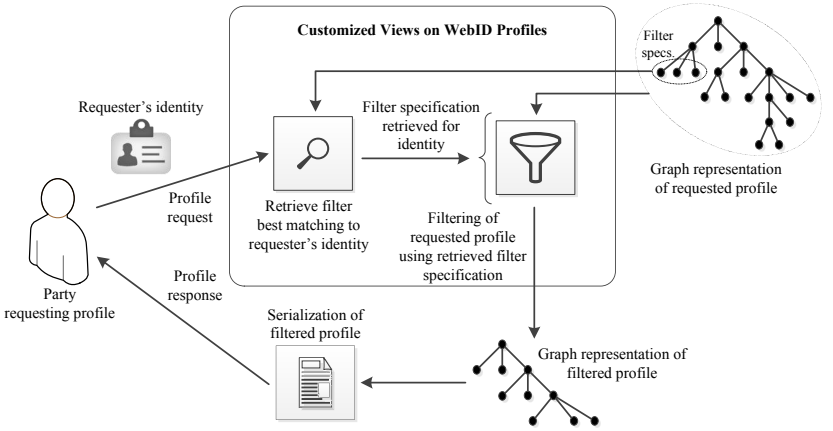

Fig. 1. Approach to Customized Views on WebID Profiles

SPARQL, the proposed *WebID Profile Filter Language (WPFL)* defines such filter specifications consisting of `filter:entity`, i.e., the requesting agent, and `filter:command`, i.e., the filter logic specified in SPARQL. WPFL also connects the filter specification with the WebID profile and is exemplarily shown below:

```
<WebID> filter:specification [
   filter:entity ENTITY;
   filter:command COMMAND ] .
```

We implement the approach using Sociddea - a WebID identity provider and management platform. Sociddea facilitates creating and editing views via a GUI, as shown in Figure 2. While the GUI targets unskilled users and is less expressive, Sociddea also allows to directly input SPARQL statements, which is more powerful but requires basic knowledge of Semantic Web technologies.

As all filter specifications are consolidated in the identity owner's WebID profile, this solution represents a portable approach. Using SPARQL as a well-established and proven language increases maintainability and flexibility, e.g., it also enables handling new identity attributes and conditional filtering.

Demonstration. In the demo session, we will show the creation WebID profiles using Sociddea. We present how users can define profile views via GUI and SPARQL queries. Finally, we demonstrate the filter selection and application depending on different requesting agents. Further information and a prototype is available at `http://vsr.informatik.tu-chemnitz.de/demo/sociddea/`.

4 Conclusion

By enabling identity owners to control the way their profile data is exposed to others, we made a significant step towards privacy in WebID-based DSNs. Filtering of WebID profile data allows identity owners to keep control about

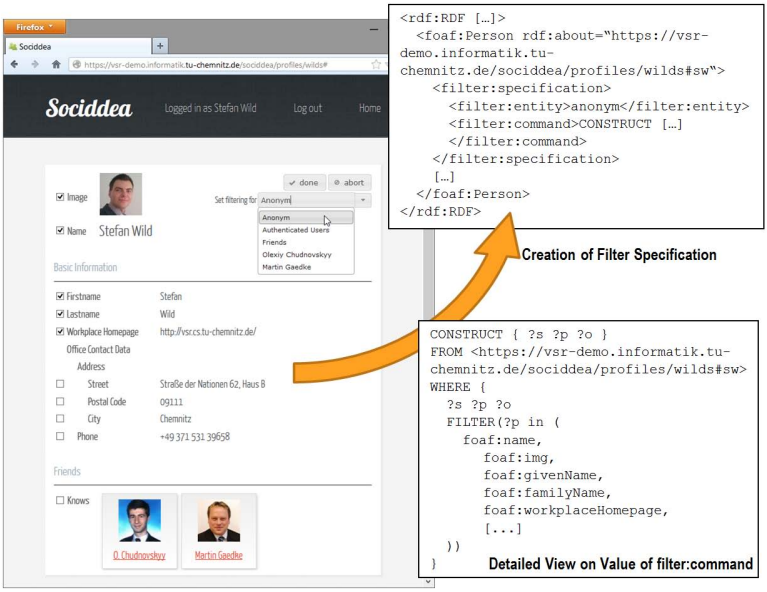

Fig. 2. Creation of Filter Specification based on User Selection

amount and nature of personal data being presented to entities requesting their profile data. In future work, we will analyze an extension of the filtering towards capabilities for dynamically adding and replacing profile data. We also plan to research the topic of reusing filters by sharing them between users of a DSN.

Acknowledgment. This work was funded by the European Commission (project OMELETTE, contract 257635).

References

1. Chudnovskyy, O., Wild, S., Gebhardt, H., Gaedke, M.: Data Portability Using WebComposition/Data Grid Service. International Journal on Advances in Internet Technology 4(3 & 4), 123–132 (2012)
2. Hackett, M., Hawkey, K.: Security, Privacy and Usability Requirements for Federated Identity (2012)
3. Tomaszuk, D., Gaedke, M., Gebhardt, H.: WebID+ACO: A distributed identification mechanism for social web (2011)
4. Tramp, S., Frischmuth, P., Ermilov, T., Shekarpour, S.: An Architecture of a Distributed Semantic Social Network. Semantic Web (2012)
5. Tramp, S., Story, H., Sambra, A., Frischmuth, P., Martin, M., Auer, S.: Extending the WebID Protocol with Access Delegation. In: Proceedings of the Third International Workshop on Consuming Linked Data (COLD 2012) (2012)
6. Yeung, C.M.A., Liccardi, I., Lu, K., Seneviratne, O., Berners-Lee, T.: Decentralization: The future of online social networking. In: W3C Workshop on the Future of Social Networking Position Papers 2 (2009)