Towards an Architecture for Pseudonymous E-Commerce

Applying Privacy by Design to Online Shopping

Sebastian Pape[1], Daniel Tasche[2], Iulia Bastys[1,3], Akos Grosz[1], Jörg Lässig[2], Kai Rannenberg[1]

Abstract: In this paper we apply privacy by design in e-commerce. We outline the requirements of a privacy-aware online shopping platform that satisfies the principle of data minimization and we suggest several architectures for building such a platform. We then compare them according to four dimensions: privacy threats, transparency, usability and compatibility with existing business models. Based on the comparison, we aim to build the selected platform in the next step.

Keywords: privacy by design; pseudonymity; data minimisation; online shopping; e-commerce

1 Introduction

E-commerce is playing an increasingly important role for operators of shopping platforms and their customers. The estimated revenue for the German e-commerce market in 2017 amounts to €55 billion, while recent statistics forecast 58 million users and a market volume of €78 billion in 2021 [St16]. Shopping platform operators are collecting customer data, as personalized offers and recommendations lead to higher revenues. Despite an increase in public's awareness on the issue of data protection and growing concerns about the usage of their data, currently e-commerce users have no alternative to disclosing personal data and revealing shopping behavior [Jo16]. A recent study reveals that 50% of online services send full information about the users' baskets to Paypal (if PayPal was selected as payment method), which in turn forwards the information, *who* purchased *what* and *where*, to a third-party specialized in data aggregation [Pr16]. At least in Europe, these issues begin to be addressed through several regulations and directives. The General Data Protection Regulation (GDPR), planned to be applied in May 2018 in the EU countries, requires data protection by design and by default: "The controller shall implement appropriate technical and organisational measures, such as *pseudonymisation*, which are designed to implement data-protection principles, such as data *minimisation* [. . .] in order to [. . .] protect the rights of data subjects" [Re16]. Therefore, we aim to improve the processes in e-commerce in respect to the data protection principles pseudonymisation and data minimisation.
The e-shopping platform could track the user's online activity through IP address, third party web-tracking [MM12], browser fingerprinting [Ec10], canvas fingerprinting [Ac14],

[1] Goethe University, Chair of Mobile Business & Multilateral Security , firstname.lastname@m-chair.de
[2] University of Applied Sciences Zittau/Görlitz, {d.tasche,j.laessig}@hszg.de
[3] Chalmers University of Technology, Gothenburg, Swedenbastys@chalmers.se

doi:10.18420/sicherheit2018_01

or evercookies [Ac14]. We do not investigate them in this paper, as previous work has already suggested different countermeasures [DMS04, PCM13, Ba13, LRB16].
Following these requirements, we make a step forward in preserving customer's privacy in online shopping, by describing an e-commerce platform that satisfies the principles of data minimization and pseudonymization (Section 3). We then suggest several architectures for building such a platform (Section 4) and compare them based on the privacy threat analysis methodology LINDDUN [WJ15], but also with respect to usability, transparency and compatibility to existing business models (Section 5).

2 Related Work

Growing concern about user traceability when making electronic payments propelled efforts in the area of privacy-preserving e-commerce. Initial work mainly concentrates on anonymous electronic payment methods through cryptographic mechanisms such as blind signatures [Ch83, Ch85, CFN90]. Aiello et al. [AIR01] describe a cryptographic protocol for anonymous shopping of digital goods based on priced oblivious-transfer and private information retrieval [Ch95]. In their setting, the customer makes an initial deposit which is later used to retrieve the desired items. Besides the initial deposit and the interaction with the platform, the online shop learns nothing else. In particular, it does not learn what or how much is purchased, nor when the buyer runs out of credit. While interesting, this approach is not feasible for deployment, as the customer would have to download the entire encrypted database. More recent work brings several improvements to the underlying protocols [RR01, CDN09, CDN10, HOG11], but they still only focus on *digital* goods, while our interest is in achieving customer privacy when purchasing *physical* goods.
A first step towards anonymous and pseudonymous e-commerce addresses the problem of purchasing goods with digital assets in a privacy-friendly manner [Sa14, GGM16, Go17]. Goldfeder et al. [Go17] introduce a series of escrow protocols to use when buying physical products online and paying with Bitcoin. While some of these protocols satisfy strong security properties, the buyer is still required to provide the seller with an address for delivering the goods, breaking to some extent buyer anonymity. Even though the seller does not learn the exact address of the buyer (as the address of a friend or of a post office can be provided instead), the seller learns the location where the product has to be dispatched.

3 System Overview

First, we give a brief overview of the involved parties and the relevant data.

Involved parties. The system consists of the following five parties:
- The User is a (registered) customer interested in purchasing goods online from Shop.
- Shop is the party that sells the (physical) goods through a platform accessible via Internet.
- The payment provider Pay collects the payment from User and transfers it to Shop.
- The logistics provider Shipping delivers the purchased goods from Shop to User.
- ID-Provider is a third-party responsible for managing the user's profile.

In order to prevent the Shop from collecting customers' private data and creating dossiers that reveal shopping behavior, we introduce a trusted third party in the system, ID-Provider, that increases the usability and the privacy of the architecture. It is responsible with managing the User's real and generated identities. A customer registers with ID-Provider with the real identity, and receives from ID-Provider a new generated identity, a pseudonym for logging in with Shop. Basically it acts as an authentication provider with pseudonymous identities, single sign on system for online shops and shopping process management system that connects the stakeholder for one shopping procedure. ID-Provider increases the usability for the User as well as the privacy of the overall shopping process. We require the user to provide the real identity in order to prevent system abuse. The pseudonym can be lifted in case of proved misbehavior. User can use the same pseudonym on multiple online platforms, or can create several pseudonyms, one for every platform, or even one for every purchase on the same platform.

User data. For a successful purchase, the user needs to provide the following information:

- *Product data* refers to the products selected by User for purchase.
- *Total value* refers to the purchasing price of the selected products plus additional payment and shipping charges to User.
- *Payment data* represents the data needed for a successful payment. Depending on the selected payment method this can be name, full address, bank account or credit card number, or even an anonymous payment method as sketched in Section 2. In general, banks and financial service providers require more information about a payment than just the bank account and the total value.
- *Delivery data* represents the information the delivery service Shipping needs for a successful delivery to User. In most cases, this is the name and address of User. However, other options are container freight stations and poste restante delivery, which do not necessarily require the same information.

The identifiability of the User and the linkability of purchases by Pay and Shipping depends on the chosen payment and delivery methods and applies to all architecture scenarios we will further discuss (Section 4). We assume that none of the parties collude, as collusion between Shop and any of ID-Provider, Pay or Shipping is sufficient for User profiling.

System requirements

A representation of a current e-shopping process is depicted in Figure 1. In general, Shop collects the User's data required for payment processing and package delivery. While it is possible to use a payment provider, such as Paypal [Pa17], and not provide Shop with any payment information, in most cases, the payment provider offers Shop a possibility to manage the payments and allows it to access the user's payment data.

As already discussed in Section 1, we ignore other customer tracking possibilities and focus our analysis on the data provided by User to the other parties. If a privacy-friendly online shopping platform would exist, the users could try to protect themselves via technical measures or legislation could protect the users by banning tracking without their consent.

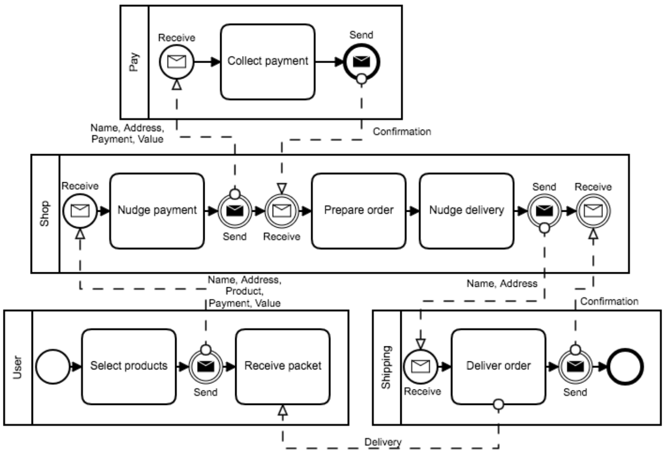


Fig. 1: Traditional Shopping Process in Business Process Modeling Notation [Ob11]

Since the login process will not differ much for the proposed architectures, the data in focus are product data, the value of the products, payment data and shipping data. When designing the pseudonymous e-commerce architecture, we aim for the *principle of minimum disclosure* under the constraints that the usability of the system should be comparable to the current systems in practice and that the process should be as transparent as possible to the user. Additionally, as discussed in Section 1, to promote a widespread use of our architecture, the shop providers business model should be respected. To chose our basic architecture, we consider the following dimensions for requirements and comparison:

Privacy. Shop should learn only the user's activity on the platform, i.e. the purchased products and their total value. The vendor does not learn payment or shipping data. Pay should learn nothing except for the amount to be payed by the user to the vendor and the payment data from the user. More specifically, the payment service does not learn the products the user purchased, but only their total value. Shipping should learn only the shipping data, but not the content (purchased products) of the package(s) to be delivered.

For the privacy analysis we apply LINDDUN, a privacy threat analysis methodology [De11, WJ15] which supports analysts in eliciting privacy requirements – similar to the security threat modeling framework STRIDE [Sh14]. Since we have a manageable number of entities in the context of our architecture scenarios, we don't run into the risk of threat explosion and can use the LINDDUN privacy analysis framework to systematically account for privacy specific threats. It is based on the graphical representation of the system's abstract representation by a data flow diagram (DFD) and the subsequent mapping of the framework's six high-level threats to each DFD element. Therefore, we model the process from checkout via payment to the delivery procedure of the products in a DFD. For each entity, we investigate the threats and map them to the elements in the DFD. In the following, we briefly discuss the privacy threats we are going to analyze.

Since the user should be able to shop pseudonymously, we consider the *identifiability* of the user (e.g. by payment or delivery data) as the main threat. In that context it is also important which parties hold which data. Depending on the party, information on *purchased products*, *the value of the purchased products*, *payment data* and *delivery data* is necessary for providing the service. As discussed, in particular the last data is suitable to revoke the User's pseudonymity. Therefore, we consider the *disclosure* of this *data* as another threat. Even if the User is not directly identifiable, *linkability* of two (or more) purchases of a user could reveal sensitive information leading to identification or at least the building of a meaningful profile. We further investigate which of the parties is able to *detect login*, *purchase*, *payment* and *delivery* events which could also be used to profile the User. Detectablility of one of the events does not mean the corresponding data is revealed, but in most cases the involved party can be identified (e.g. User did payment with Pay but neither amount nor payment data can be seen). *Unawareness* and *non-compliance* are out of the scope, as they are more related to the user interface and the entities' policies which are independent of our system architecture process. We are also not regarding *non-repudiation* for this paper since we consider it more related to contracting and legal aspects than to the architecture of our shopping platform.

Usability. Many aspects concerning the usability will not depend on the system's architecture, but on proper user interfaces allowing the user to manage his data in a easy and transparent way. However, in order to allow the user to easily use the system from different clients (e.g. computer, tablet, smart phone, . . .), the user should not store information such as a cryptographic key. Additionally, the speed of the system should be comparable to existing systems, thus complex cryptographic protocols which delay the process too much can not be used. As a consequence, certain privacy enhancing technologies such as attribute based credentials [SKR12] do not come into play, because they make use of cryptographic keys, which the user would have to store on a smartcard. We compare the different architectures based on the effort the user needs to take for.

Transparency. A natural data flow which allows the user to easily understand which data is provided to whom for which purpose contributes to a transparent system. Since the user interface is out of the scope of this work, transparency of the different architectures will depend only on data flows.

Compatibility to existing business models. Analogous to attribute based credentials [Sa15], we assume that when preserving the online shop providers' business models, a broad distribution of our platform can be more easily achieved. Certainly, this does not mean that the shop providers should be allowed to collect all data they want. But allowing them to keep profiles for pseudonyms and sending e.g. newsletters (via ID-Provider) to users who gave consent would certainly be helpful for the adoption of pseudonymous e-commerce.

4 Architecture Variants

In this section we describe the three architectures, we considered for implementation. For an easier comparison, we also analyzed the current shopping process. The standard architecture allows the Shop to gather a big volume of data about its users. In order to avoid this, we suggest three architectures, two of them make use of public-key infrastructure (see Sections 4.2 and 4.3) and a third one without encryption but self data hosting (see Section 4.4). All scenarios involve an ID-Provider for managing the user's profile.

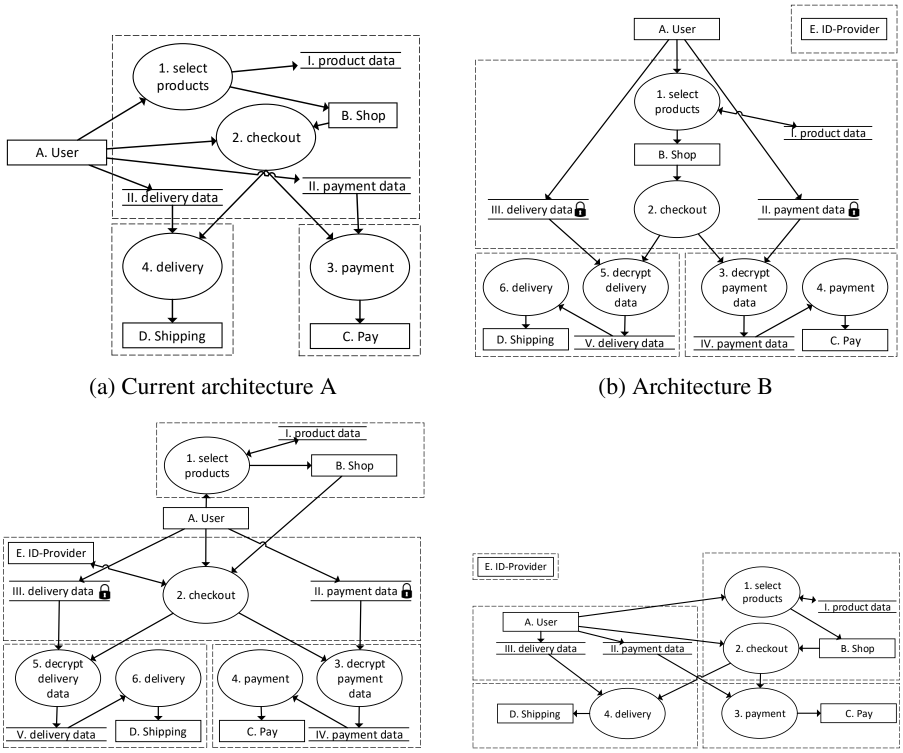
For the following analysis, we abstract from the login process and from confirmations as far as possible. Although other variants exists, we assume the User selects the products, pays and gets them delivered afterwards. Special care has to taken that Pay and Shipping providers do not pass the User's data to the Shop, e.g. by offering an administrative user interface, where payment data is listed or sending tracking information of the delivered packages. Each architecture's description follows the following template: We describe the process of every scenario and briefly discuss advantages and disadvantages. The corresponding data flow from selecting the products, checkout, payment and delivery process is depicted in Fig. 2. The analyzed privacy threats described in Section 3 are listed in Tab. 1. For each privacy threat (from Sect. 3) we denote the scenarios where it exists. For some of the analyzed threats, it depends on the users. If users don't want the shop to link their payments, they can use a new pseudonym for each purchase. For payment and shipping it depends on the kind of service the user chooses. Clearly, it makes a difference whether they are paying with anonymous electronic payment or by providing their credit card data. For shipping they could ask for home delivery or use a container freight station. We denote these threats in brackets in Tab. 1.

4.1 A: Current Shopping Process

The standard shopping process is depicted in BPMN in Figure 1 and has already been described. Figure 2a shows the data flow diagram. The Shop collects all information about the user, and thus can identify the user and can link all shopping activities. The identifiability of the User and the linkability of purchases by Pay and Shipping depends on the chosen payment or delivery method. The highest privacy threat for the User is the Shop because of the possibility to disclose the User's payment and delivery data as well as profiling the User.

4.2 B: Shop Stores Encrypted Data

In this scenario, the user reveals only his real identity (name) to ID-Provider when registering. The ID-Provider acts as single-sign-on login service, to allow the user to log in several Shops without further registration. Additionally, ID-Provider provides public keys for payment and shipping provider. Shipping and payment data is stored encrypted on the Shop's server. The data flow of this scenario is depicted in Figure 2b.
The User initiates the process by *select products*. The Shop gets the product data and stores it. In the *checkout* process, the User decides on a payment and shipping provider. The user gets the public keys for any provider he wants to use, encrypts the payment respectively delivery data and sends it to the Shop. Subsequently, the Shop initiates the *payment* process by forwarding the encrypted banking details along with the amount to be payed to Pay. After successfully decrypting the payment data and completing the payment transaction,



(a) Current architecture A

(b) Architecture B

(c) Architecture C

(d) Architecture D

Fig. 2: Data flow diagram

Pay sends a confirmation of payment to Shop. Upon confirmation, Shop starts the *delivery* process and sends the package labeled with the User's encrypted address to Shipping. After successfully decrypting the delivery data, Shipping proceeds with delivering the package and provides Shop with a confirmation of delivery. In this architecture Shop is not able to identify the user and can not disclose payment and delivery data. Since there is a possibility to use one pseudonym for each shopping process, the shop is also not able to link the purchases of an user unless the User allows it. The ID-Provider only knows the real identity of the user, but can not disclose payment or delivery information and also does not learn anything about the purchase process. However, the ID-Provider is able to detect the login process. The information distribution of Pay and Shipping are not affected. Therefore, the same conditions apply as for the standard architecture.

Advantages and disadvantages.

+ ID-Provider does not learn methods User uses for payment.
+ The Pay and Shipping services do not learn the virtual identity of User.
– The ID-Provider is able to detect logins in the store.
– Key management: It is difficult for the User to encrypt the payment and delivery data.

Tab. 1: Privacy Threats Mapped to Architecture Variants from Sect. 4

Entity Threat	Shop	Pay	Ship	Identity Provider
Identifiability	A	(ABCD)[2]	(ABCD)[3]	B C D
Disclosure shopping cart	A B C D			
Disclosure total value	A B C D	A B C D		
Disclosure payment data	A	A B C D		
Disclosure delivery data	A		A B C D	
Linkability purchase	A(BCD)[1]	(ABCD)[2]	(ABCD)[3]	C
Detectablility login	A B C D			B C D
Detectablility purchase	A B C D	A B C D	A B C D	C
Detectablility payment	A B C D	A B C D		C
Detectablility delivery	A B C D		A B C D	C

[1] Depends on the user's choice. [2] Depends on user's payment [3] Depends on user's shipping

Either this is done in the browser (e.g. with Javascript) or by an App, but the user has to trust the party providing the code.

4.3 C: ID-Provider Stores Encrypted Data

This architecture is similar to architecture B. The only change is that the (encrypted) payment and delivery data is stored at the ID-Provider. As a consequence, instead of directly delivering the data to Pay and Shipping, the Shop refers them to the ID-Provider where they need to authenticate and ask for the User's data. Therefore, the data flow itself is very similar to the one of architecture B (see 4.2) as depicted in Figure 2c. In this scenario, ID-Provider controls the shopping process. It knows the identity of the user and has information about where the user shops but does neither know the payment or delivery data since they are encrypted nor any details of the shopping content. Shop does not know the real identity of the user nor the payment or delivery details. The data distribution or possible disclosure from Pay and Shipping are unchanged.

Advantages and disadvantages.

+ ID-Provider does not learn the payment or delivery data of the User.
– If the User does not perform the encryption himself, then he has to trust ID-Provider to provide him with the correct public keys of the payment and logistics services.
– ID-Provider learns the Shop where the User makes his purchases.
– One more point of failure: ID-Provider is involved in multiple transactions.

4.4 D: User Gets Redirected to 3rd Parties

In this scenario, ID-Provider solely acts as a single-sign-on service and certification authority. All the information required for each of the steps of the pseudonymised shopping process is stored by the User. He initiates the process by *selecting the products*. The Shop gets the product data and stores it. In the *checkout* process, the User decides on a payment and shipping provider. Subsequently, the Shop redirects the User for the *payment* process and the User delivers his payment data directly to Pay. Pay sends a confirmation of payment back to Shop. Upon confirmation, the Shop redirects the User for the *delivery* process and the User delivers his delivery data directly to Shipping. Shipping receives the package from Shop with an identifier to link it to the address and proceeds with delivering the package and provides Shop with a confirmation of delivery. Figure 2d shows the data flow. Since the User has full control about his profile data, Shop does neither know the User's identity nor the payment or delivery data. The information distribution of Pay and Shipping are not affected. Therefore, the same conditions apply as for the standard scenario.

Advantages and disadvantages.

+ The User is fully in control of his personal information.
+ Only necessary data is provided to each party.
+ Only communication needs to be encrypted.
– Additional tools have to be provided for Users to host their information.
– A lot of transactional load is put on the User. In particular, the User has to check that he is providing the information to the correct party, e.g. by checking cryptographic certificates.
– The payment process has to work instantly, otherwise additional communication is needed to synchronise payment with shipping processes.

5 Architecture comparison

In this section we compare the previously described architectures on the four dimensions described in Section 3: privacy, usability, transparency, and compatibility.

Privacy. Every involved party should learn only information about the activity belonging to its area of responsibility. In the standard scenario the Shop holds every information about the User's identity. As described in Sect. 4, all proposed architectures consider the principle of minimum disclosure. They differ only in the information provided to the ID-Provider.
In each scenario the User has the possibility to create several shopping pseudonyms. If he uses one for every shopping process, the store could not link several purchases. This applies to all architectures. The linkability of the purchase and identifiablity of the User on Pay's and Shipping's side depends on the payment and shipping methods and is not an architectural aspect. ID-Provider could link the purchases in Scenario C because ID-Provider manages the checkout process. In the other scenarios ID-Provider acts as a real identity provider and just manages the login process. Therefore, the detectability of a purchase, a payment and a delivery applies to Scenario C, but not to Scenario B and Scenario D.

Usability. Every architecture has the registration at ID-Provider and the managing of pseudonyms in common. That means compared to the standard scenario one has to maintain data not on Shop's side but on ID-Provider's side. As a compensation for managing the profiles, the User would not need to register at any Shop anymore.
Architecture B and C come with additional effort since the Users have to encrypt their data. In particular, in architecture B, Users face the problem that they might not want to

trust the Shop's App or Javascript-code making it difficult to encrypt. On the other hand in architecture C, the user has to register at the ID-Provider anyway and it seems reasonable to rely, e.g. on an App or Javascript-code on a web page. Architecture D asks the user to provide his payment and delivery data for each purchase again. This could be mitigated by making use of the The PaymentRequest API [Ba17]. However, since the recommendation is quite new, it will take some time until this has been adapted. For the authentication and single sign-on the X.509 standard could be used but needs some extensions to provide special user information. Therefore, Dash et al. [Da17] show an architecture proposal for an identity management architecture as a service. Additionally, since the Shop redirects the User to Pay and Shipping, the User has to check for each of the providers that Shop was directing her to the correct entity and not to a forged one to get the User's data.

Transparency. Despite sharing payment and delivery data directly with the Shop the standard architecture is quite transparent, because the User should be aware of sharing this data with the Shop. Although, the user might not be aware that this information might be shared with or is accessible by 3rd party service providers (e.g. webhoster, payment provider). The same holds for architecture D, where the Users need to provide their data to each entity directly. Architectures B and C, lack a bit of transparency, because it is harder for the user to assess how and from whom the encrypted data will be processed. However, it's up to the respective entity to inform the User in a supporting way.

Compatibility. The basic business processes of the involved parties are not broken by this architectures. However, by not disclosing the User's identity and therefore contact information to Shop, Shop needs to rely on ID-Provider to forward e.g. newsletters or special offers to the User. In case of misuse or disputes, ID-Provider is needed to reveal the User's identity. Pay and Shipping need to adjust their processes, in order to not reveal the User's data to Shop. However, there is no large difference here between architectures B, C, and D.

Final Architecture. Table 2 shows an overview of all attributes concerning the four analyzed architectures. While architectures B and D are favorable in respect to privacy and transparency, our focus when defining the requirements was to put emphasis on usability. Improved privacy should not complicate the shopping process for the user. The slight disadvantage in transparency from architecture C to D does not outweigh the disadvantage of architecture D that Users need to provide their data for each purchase again or alternatively have additional accounts (and logins) at payment and shipping providers. Therefore, we believe architecture C to be the most feasible option.

	Privacy	Usability	Transparency	Compatibility
A	-	o	+	++
B	++	+	o	+
C	+	++	o	+
D	++	o	+	+

Tab. 2: Comparison of the several architectures.

6 Conclusion and Future Work

In the context of pseudonymous online shopping, we presented and assessed three different architectures and compared them to the existing architecture. So far, the proof of concept shows, that a pseudonymous e-commerce process can be set up in a usable and privacy-friendly way. The User data is no longer on Shop's side but split to several parties that are involved in the shopping process.

We plan to add more processes to the shopping system such as returning goods and writing invoices. Around this, several legal and technical issues need to be resolved, e.g. how the Shop can issue an invoice to a pseudonym. Even though the PaymentRequest API only supports non-normative encryption of data fields and might also expose payments methods (cf. [Ba17, Section 19.2]), it might be helpful in storing payment and delivery data in the User's computer to avoid creating a centralized database.

Future work includes the implementation of certain restrictions for Users. For example, only Users above certain age or in certain geographical regions can access certain products. The next steps also contain the detailed description of the used protocol.

7 Acknowledgments

The SIOC project [SI] is supported by the German Federal Ministry of Education and Research's (BMBF) program "Datenschutz: selbstbestimmt in der digitalen Welt".

References

[Ac14] Acar, Gunes; Eubank, Christian; Englehardt, Steven; Juarez, Marc; Narayanan, Arvind; Diaz, Claudia: The web never forgets: Persistent tracking mechanisms in the wild. In: CCS. 2014.

[AIR01] Aiello, William; Ishai, Yuval; Reingold, Omer: Priced Oblivious Transfer: How to Sell Digital Goods. In: EUROCRYPT. 2001.

[Ba13] Bau, Jason; Mayer, Jonathan; Paskov, Hristo; Mitchell, John C: A promising direction for web tracking countermeasures. W2SP, 2013.

[Ba17] Bateman, Adrian; Koch, Zach; McElmurry, Roy; Denicola, Domenic; Cáceres, Marcos: , Payment Request API. https://www.w3.org/TR/2017/CR-payment-request-20170921/, 2017. W3C Candidate Recommendation 21 September 2017.

[CDN09] Camenisch, Jan; Dubovitskaya, Maria; Neven, Gregory: Oblivious transfer with access control. In: CCS. 2009.

[CDN10] Camenisch, Jan; Dubovitskaya, Maria; Neven, Gregory: Unlinkable priced oblivious transfer with rechargeable wallets. In: FC. 2010.

[CFN90] Chaum, David; Fiat, Amos; Naor, Moni: Untraceable electronic cash. In: CRYPTO. 1990.

[Ch83] Chaum, David: Blind signatures for untraceable payments. In: CRYPTO. 1983.

[Ch85] Chaum, David: Security without identification: Transaction systems to make big brother obsolete. CACM, 1985.

[Ch95] Chor, Benny; Goldreich, Oded; Kushilevitz, Eyal; Sudan, Madhu: Private information retrieval. In: FOCS. 1995.

[Da17] Dash, Pritam; Rabensteiner, Christof; Hörandner, Felix; Roth, Simon: Towards Privacy-Preserving and User-Centric Identity Management as a Service. In (Fritsch, Lothar;

Roßnagel, Heiko; Hühnlein, Detlef, eds): Open Identity Summit 2017. Gesellschaft für Informatik, Bonn, pp. 105–116, 2017.

[De11] Deng, Mina; Wuyts, Kim; Scandariato, Riccardo; Preneel, Bart; Joosen, Wouter: A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. Requirements Engineering, 2011.

[DMS04] Dingledine, Roger; Mathewson, Nick; Syverson, Paul: Tor: The second-generation onion router. Technical report, Naval Research Lab Washington DC, 2004.

[Ec10] Eckersley, Peter: How unique is your web browser? In: PETS. 2010.

[GGM16] Garman, Christina; Green, Matthew; Miers, Ian: Accountable privacy for decentralized anonymous payments. In: FC. 2016.

[Go17] Goldfeder, Steven; Bonneau, Joseph; Gennaro, Rosario; Narayanan, Arvind: Escrow protocols for cryptocurrencies: How to buy physical goods using Bitcoin. 2017.

[HOG11] Henry, Ryan; Olumofin, Femi; Goldberg, Ian: Practical PIR for electronic commerce. In: CCS. 2011.

[Jo16] Jourova, Vera: , How does the data protection reform strengthen citiziens' rights? http://ec.europa.eu/justice/data-protection/document/review2012/factsheets/factsheet_dp_reform_citizens_rights_2016_en.pdf, 2016.

[LRB16] Laperdrix, Pierre; Rudametkin, Walter; Baudry, Benoit: Beauty and the beast: Diverting modern web browsers to build unique browser fingerprints. In: IEEE S&P. 2016.

[MM12] Mayer, Jonathan R; Mitchell, John C: Third-party web tracking: Policy and technology. In: IEEE S&P. pp. 413–427, 2012.

[Ob11] Object Management Group: , Notation (BPMN) version 2.0. OMG Specification, 2011.

[Pa17] Paypal: , Paypal Website. https://www.paypal.com, 2017.

[PCM13] Perry, Mike; Clark, Erinn; Murdoch, Steven: The design and implementation of the Tor Browser. Technical report, The Tor Project, 2013. https://www.torproject.org/projects/torbrowser/design/.

[Pr16] Preibusch, Sören; Peetz, Thomas; Acar, Gunes; Berendt, Bettina: Shopping for privacy: Purchase details leaked to PayPal. Electronic Commerce Research and Applications, 2016.

[Re16] Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance). Official Journal of the European Union, L 119/1, http://data.europa.eu/eli/reg/2016/679/oj, 2016.

[RR01] Ray, Indrakshi; Ray, Indrajit: An Anomymous Fair Exchange E-commerce Protocol. In: IPDPS. 2001.

[Sa14] Sasson, Eli Ben; Chiesa, Alessandro; Garman, Christina; Green, Matthew; Miers, Ian; Tromer, Eran; Virza, Madars: Zerocash: Decentralized anonymous payments from bitcoin. In: IEEE S&P. 2014.

[Sa15] Sabouri, Ahmad: Understanding the Determinants of Privacy-ABC Technologies Adoption by Service Providers. In: Open and Big Data Management and Innovation : 14th IFIP WG 6.11 Conference on e-Business, e-Services, and e-Society, I3E 2015. 2015.

[Sh14] Shostack, Adam: Threat modeling: Designing for security. 2014.

[SI] SIOC project website. https://sioc.eu/.

[SKR12] Sabouri, Ahmad; Krontiris, Ioannis; Rannenberg, Kai: Attribute-Based Credentials for Trust (ABC4Trust). In: TrustBus. 2012.

[St16] Statista: , E-Commerce in Deutschland. https://de.statista.com/outlook/243/137/ecommerce/deutschland/, 2016.

[WJ15] Wuyts, Kim; Joosen, Wouter: LINDDUN privacy threat modeling: a tutorial. 2015.

All websites have been last accessed on Dec. 11th, 2017.