Available online at www.sciencedirect.com

SciVerse ScienceDirect

www.compseconline.com/publications/prodclaw.htm

Computer Law & Security Review

Open-source intelligence and privacy by design

Bert-Jaap Koops [a], Jaap-Henk Hoepman [b,c], Ronald Leenes [a]

[a] TILT – Tilburg Institute for Law, Technology, and Society, Tilburg University, PO Box 90153, 5000 LE Tilburg, The Netherlands
[b] TNO, The Netherlands
[c] Radboud University Nijmegen, The Netherlands

Keywords:
OSINT
Open sources
Social networking
Privacy by design
Techno-regulation
Policy mark-up language
Revocable privacy

ABSTRACT

As demonstrated by other papers on this issue, open-source intelligence (OSINT) by state authorities poses challenges for privacy protection and intellectual-property enforcement. A possible strategy to address these challenges is to adapt the design of OSINT tools to embed normative requirements, in particular legal requirements. The experience of the VIRTUOSO platform will be used to illustrate this strategy. Ideally, the technical development process of OSINT tools is combined with legal and ethical safeguards in such a way that the resulting products have a legally compliant design, are acceptable within society (social embedding), and at the same time meet in a sufficiently flexible way the varying requirements of different end-user groups. This paper uses the analytic framework of privacy design strategies (minimise, separate, aggregate, hide, inform, control, enforce, and demonstrate), arguing that two approaches for embedding legal compliance seem promising to explore in particular. One approach is the concept of revocable privacy with spread responsibility. The other approach uses a policy mark-up language to define Enterprise Privacy Policies, which determine appropriate data handling.

Both approaches are tested against three requirements that seem particularly suitable for a 'compliance by design' approach in OSINT: purpose specification; collection and use limitation and data minimisation; and data quality (up-to-dateness). For each requirement, the paper analyses whether and to what extent the approach could work to build in the requirement in the system. The paper concludes that legal requirements cannot be embedded fully in OSINT systems. However, it is possible to embed functionalities that facilitate compliance in allowing end-users to determine to what extent they adopt a 'privacy-by-design' approach when procuring an OSINT platform, extending it with plug-ins, and fine-tuning it to their needs. The paper argues that developers of OSINT platforms and networks have a responsibility to make sure that end-users are enabled to use privacy by design, by allowing functionalities such as revocable privacy and a policy-enforcement language.

0267-3649/$ – see front matter © 2013 Bert-Jaap Koops, Jaap-Henk Hoepman and Ronald Leenes. Published by Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.clsr.2013.09.005

1. Introduction[1]

Open-source intelligence (OSINT) involves the collection, analysis, and use of data from open sources for intelligence purposes. Twitter feeds and Facebook pages are for instance mined for law enforcement purposes and online (streaming) news channels are monitored for information that may be relevant to prevent and detect terrorist activity. The fact that data are openly available does not mean that they can be processed without regard to legal issues: certain legal requirements apply to open-source data. In particular, as demonstrated by other papers in this issue, OSINT by state authorities poses challenges for privacy protection and intellectual-property enforcement. For instance, although the data concerned may be available to everyone, it still may concern privacy sensitive information and also the use of publicly available audio and video may constitute copyright infringements. A possible strategy to address these challenges is to adapt the design of OSINT tools to embed normative requirements, in particular legal requirements. Ideally, the technical development process of OSINT tools is combined with legal and ethical safeguards in such a way that the resulting products have a legally compliant design, are acceptable within society (social embedding), and at the same time meet in a sufficiently flexible way the varying requirements of different end-user groups.

This paper aims to investigate how and to what extent legal requirements can be safeguarded in the design of OSINT systems. To keep the analysis feasible, we will limit ourselves to privacy requirements and thus focus on privacy by design. We will focus on OSINT platforms as these are currently being developed rather than on specific OSINT tools, as the platforms aim at providing an interoperable forum for integrating diverse OSINT software and hardware. Since a privacy-by-design approach to OSINT tools is dependent on, among other things, the platform these tools will be plugged into, it is crucial for the viability of OSINT privacy by design that the platforms are compatible with privacy-by-design features. We will use the OSINT platform as developed in the European VIRTUOSO project as an illustration.[2]

The paper is structured as follows. We start with outlining the idea of technology as a regulatory tool – techno-regulation – and the notion of privacy by design (Section 2). We then zoom in on privacy-by-design models, using the analytic framework of privacy design strategies (minimise, separate, aggregate, hide, inform, control, enforce and demonstrate).[3] We argue that two approaches for embedding legal compliance seem promising to explore in particular: the concept of revocable privacy with spread responsibility, and policy mark-up language to define Enterprise Privacy Policies (Section 3). Both approaches are subsequently tested against three requirements that seem particularly suitable for a 'compliance by design' approach in OSINT: purpose specification; collection and use limitation and data minimisation; and data quality (up-to-dateness). For each requirement, the paper analyses whether and to what extent the approach could work to build in the requirement in the system (Section 4). We end with a brief summary and conclusion (Section 5).

2. Background: techno-regulation and privacy by design

Technology is an instrument that is or can be used to achieve regulation, i.e., the attempt to alter the behaviour of others to standards or goals with the intention of producing a broadly identified outcome.[4] Using technology as a regulatory instrument is often termed, following Lessig, 'code as law' or 'code as code'.[5] A synonym introduced by Roger Brownsword is 'techno-regulation'.[6] All of these can be defined as the 'deliberate employment of technology to regulate human behaviour'.[7]

With the rise of cyberspace, information and communication technologies became an instrument of regulation, and the importance of the normative effects of technology and its relation with law were firmly planted on the agenda. Lessig argues that regulation in cyberspace increasingly is shaped by technology, rather than by law[8]: 'In cyberspace we must understand how code regulates – how software and hardware that make cyberspace what it is regulate cyberspace as it is. (…) [T]his code is cyberspace's "law." *Code is law*.'[9]

Thus, techno-norms can be seen as instruments that regulate behaviour. Regulatory actors intentionally employ rules embedded into the technology to achieve certain (policy) goals. The embedded norms constitute the space of possible actions that users can perform; in that sense, they are norm setting. At the same time, the technology also enforces, to a lesser or greater extent, to what extent the norm will be followed. Thus, in techno-regulation, norm setting and norm enforcement coincide. Technology in this sense presents an ideal tool to regulate the behaviour of individuals, especially in the Internet context. From the perspective that techno-norms affect the behaviour of individuals and can restrict their autonomy and freedom to act, the use of techno-

[1] This article is based on research conducted in the VIRTUOSO project, in particular on the report B.J. Koops et al., *D3.4 Code As Code Assessment*, 27 April 2012, available at http://www.virtuoso.eu under 'Documentation'.

[2] http://www.virtuoso.eu.

[3] J.-H. Hoepman, *Privacy Design Strategies*, October 2012. eprint arXiv:1210.6621.

[4] Black, J. (2005), 'What is Regulatory Innovation?' in: Black, J., Lodge, M. Thatcher, M. (eds), *Regulatory Innovation*, Edward Elgar, pp. 103–146.

[5] Lessig, L. (1999), *Code and other laws of cyberspace*. New York, Basic Books.

[6] Brownsword, R. (2004), 'What the World Needs Now: Techno-Regulation, Human Rights and Human Dignity'. in: Global Governance and the Quest for Justice (edited by Brownsword, R.). 4: Human Rights. Hart, pp. 203–234.

[7] Leenes, R.E. (2010), *Harde Lessen: Apologie van technologie als reguleringsinstrument*, Tilburg: Universiteit van Tilburg, p. 21.

[8] This is not to say that the Internet or cyberspace is *only* regulated through technology; Lessig's statement is used here to emphasise that cyberspace is *also*, and perhaps increasingly, regulated through technology besides or on top of legal regulation.

[9] Lessig (n 5) (emphasis in original).

regulation requires justification, and this applies equally to restrictions imposed by the state and to those imposed by private entities. In the context of this paper, the legitimation lies in the fact that the techno-regulation aims to ensure compliance with legal norms, and can thus serve to protect legal subjects. In this sense, using a techno-regulatory approach in OSINT platforms is not techno-regulation in the sense of constituting new norms to restrict the behaviour of end-users, but as techno-regulation in the sense of technically-enforced legal compliance.

In relation to privacy, techno-regulation takes the form of Privacy-Enhancing Technologies (PETs), which can be defined as 'a coherent system of ICT measures that protects privacy by eliminating or reducing personal data or by preventing unnecessary and/or undesired processing of personal data, all without losing the functionality of the information system'.[10] The idea of privacy by design has been widely embraced in European policy and in proposed legislation, stressing the need that information systems be designed in such a way that privacy and data protection rules are automatically enforced and that default settings restrict data processing to a necessary minimum.[11,12] So far, however, PETs have not been widely deployed in practice, partly through a lack of incentives, and perhaps because of conceptual difficulties of translating flexible legal norms into more rigid technology-embedded rules. There are, however, many technical solutions that can partly help to protect privacy, if they are part of a wider and integrated strategy of privacy by design. Such an approach might benefit from privacy design strategies, which can help bridge the gap between the abstract notion of privacy by design and the concrete tools of privacy-enhancing technologies. This will be elaborated in the next section.

3. Models and technologies for privacy by design

3.1. *Introduction*

Data protection law regulates the processing of personal data, i.e., data that can be linked, with reasonable effort, to a natural person. A reasonable effort in this context is for example to combine data from several different databases (even if some of those are not under someone's immediate control) to establish such a link. The goals of technical measures to protect privacy are to make it difficult (if not practically impossible) to link a piece of information to a natural person, to limit the processing of personal data to defined uses, and to give users control over their personal data once their data are disclosed. Much research regarding PETs has focused on obfuscating or hiding the link between persons and their personal data. These technologies aim to achieve a certain level of anonymity, unlinkability or unobservability.[13] Less research has addressed controlling data once it has been released, although there are efforts to limit the processing of personal data based on stated data handling and data access policies.[14]

A primary way to comply with data protection law is to make sure that no personal data are (further) processed by anonymising personal data. Anonymity can be defined as the state of being not identifiable within a set of subjects, which captures the intuitive notion of 'hiding in the crowd'. Unlinkability guarantees that two events or data items cannot be linked to each other. Examples of such events are visiting several websites, or sending several emails. Examples of such data items are one's subscription to a newspaper, or one's current place of work. Finally, unobservability guarantees that nobody is able to tell whether a certain event (like sending a message) did or did not take place.[15]

As indicated, privacy by design refers to the underlying philosophy of protecting privacy in the early design stage of technological development. In the context of the developing IT systems, this implies that privacy protection is a system requirement that must be treated like any other functional requirement. As a result, also privacy protection will have an impact on the design and implementation of the system.

To support privacy by design, we therefore need guiding principles for these system development phases. There are general principles that can guide system architecture without imposing a specific structural organisation or schema for the system. We will refer to such higher level abstractions that structure the system architecture in terms of *design strategies*. Privacy by design starts from a set of abstract privacy principles that capture the essence of protecting informational privacy. The principles can be effected using various design patterns; a 'design pattern' is a concept that, at a high level of abstraction,

> *provides a scheme for refining the subsystems or components of a software system, or the relationships between them. It describes commonly recurring structures of communicating components that solves a general design problem within a particular context.*[16]

[10] European Commission (2007), *Communication on Promoting Data Protection by Privacy Enhancing Technologies (PETs)*. Brussels: EC, 2 May 2007.

[11] In a strict sense this is 'data protection by design', which is a part of the broader 'privacy by design', and is restricted to informational privacy.

[12] Privacy by Design Resolution at the 32nd International Conference of Data Protection and Privacy Commissioners, Jerusalem, 27–29 October 2010. Available online at http://www.privacyconference2011.org/htmls/adoptedResolutions/2010_Jerusalem/2010_J5.pdf; Article 29 Working Party (2009), *The Future of Privacy*. Brussels: Article 29 Data Protection Working Party, 1 December 2009; Art. 23 Proposal for a Regulation on the protection of individuals with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation), COM (2012) 11final, 25.01.2012.

[13] Pfitzmann, A., & Hansen, M. (2010), Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management – a consolidated proposal for terminology. http://dud.inf.tu-dresden.de/Anon_Terminology.shtml.

[14] See, for instance, J. Camenisch, S. Fischer-Hübner, & K. Rannenberg (Eds.) (2011), *Privacy and identity management for life*. Dordrecht|Heidelberg: Springer.

[15] *Pfitzmann & Hansen (n 13)*.

[16] Buschmann, F., R. Meunier, H. Rohnert, P.Sommerlad, and M. Stal (1996), *Pattern-Oriented Software Architecture: A System of Patterns*, John Wiley and Sons.

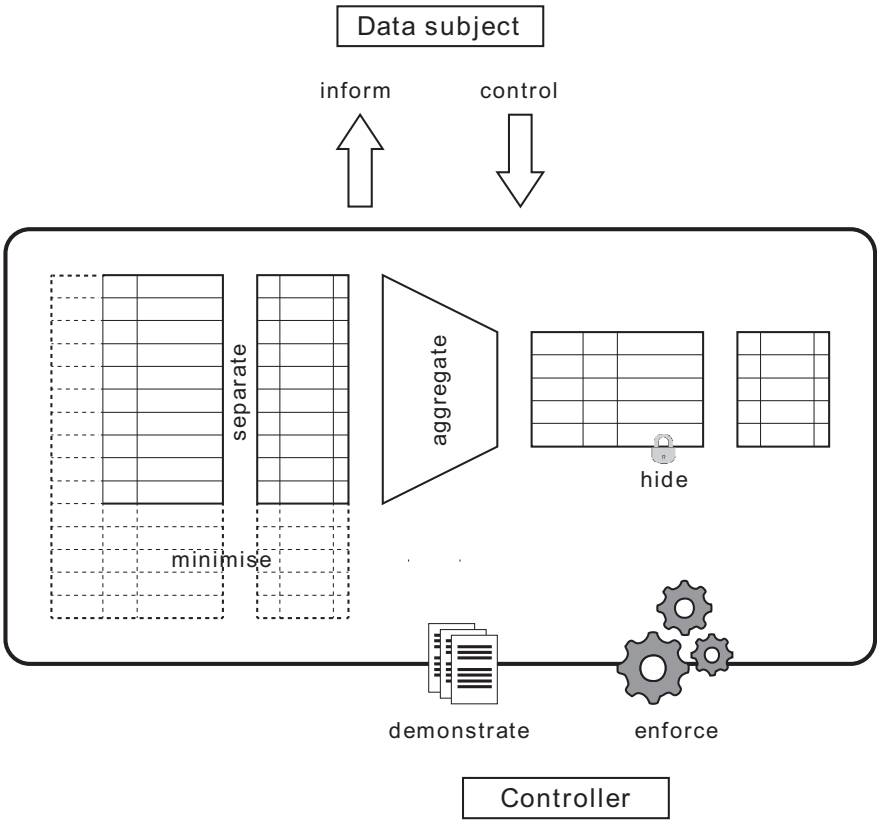

Fig. 1 – Privacy design principles.

In the following sections we will first discuss the general privacy-by-design strategies and show which design patterns can be applied to make them more concrete. Subsequently, we will discuss some major PETs that can be used to implement these patterns in real-life systems, as well as technical measures that focus on legal compliance.

3.2. *Privacy design strategies*

We distinguish the following eight privacy design principles: minimise, separate, aggregate, hide, inform, control, enforce and demonstrate.[17] A graphical representation of these principles, when applied to a database system, is given in Fig. 1. In this representation, the privacy principles are applied to tables in a database. In a database context, *minimising* corresponds to not storing certain rows or columns in a database, *separate* means splitting data over several tables, and *aggregate* implies storing data at an aggregate level. *Hide* means that personal data and their interrelationships are hidden from plain view. *Inform* and *demonstrate* provide transparency of the system's behaviour. *Control* allows the user to control the data processing. *Enforce* refers to data handling policies to be observed by the system. In the following subsections we will elaborate the eight privacy design strategies, in relation to the collection, handling, storage, and dissemination of personal data.

3.2.1. *Minimise*

The most basic privacy design strategy is data minimisation: *The amount of personal data that is processed should be minimal*. By ensuring that only essential data is collected, the possible privacy impact of a system is limited. Data minimisation can take two forms: either a yes/no decision to collect any data about individuals is made (as a consequence, for some people no information will be collected at all), or the amount of data that is collected about each individual is restricted to a limited set of characteristics.

Common design patterns implementing this strategy are 'select before you collect',[18] anonymisation and the use of pseudonyms.[19]

3.2.2. *Separate*

The second design strategy is data or process separation: *The processing of personal data should be done in a distributed fashion whenever possible*. By separating the processing or storage of several sources of personal data that belong to the same person, complete profiles of one person cannot be easily made.

The principle of separation calls for distributed processing instead of centralised solutions. In particular, data from separate sources should be stored in separate databases, and these databases should not be linked if not needed. Data

[17] For a more extensive discussion, see J.-H. Hoepman, *Privacy Design Strategies*, October 2012. eprint arXiv:1210.6621.

[18] Jacobs, B. (2005). 'Select before you collect', 54 *Ars Aequi*, pp. 1006–1009.

[19] Pfitzmann and Hansen (n 13).

should be processed locally whenever possible, and stored locally if feasible as well. Database tables should be split when possible (and links between rows should be hard to find).

3.2.3. *Aggregate*

The third design strategy, aggregate, states: Personal data should be processed at the highest level of aggregation and with the least possible detail in which they are (still) useful. By restricting the amount of detail, or by considering data at the group level instead of considering data for each person separately, the personal data become less sensitive. When the data is sufficiently coarse-grained, and the size of the group over which they are aggregated is sufficiently large, little information can be attributed to single persons, thus protecting their privacy.

Common design patterns are aggregation over time, location granularity, and group-level profiles.

3.2.4. *Hide*

The fourth design strategy, hide, holds: *Any personal data, and their interrelationships, should be hidden from plain view*. By hiding personal data from plain view, it cannot easily be abused. Depending on the context, the data may be hidden from third parties (i.e. that are not the controller or a legitimate processor), or it may be hidden from any party at all (including the data controller or a processor). Information that spontaneously emerges from the use of a system (for example data from various sources that through their combination can be linked to a unique individual), should be hidden from everybody. Information that is collected, stored or processed legitimately by one party should be hidden from any other, third, party. In this case, the principle corresponds to ensuring confidentiality.

Common design patterns are the use of encryption (locally, or on the network using SSL or other security protocols), or the use of onion routing to hide traffic patterns.

3.2.5. *Inform*

The *inform* strategy corresponds to the important notion of transparency: *Data subjects should be adequately informed whenever personal data is processed*. Data protection regulation requires that data subjects are properly informed about the fact that personal data is processed when they use a certain system.[20] Data subjects should be informed about which data are processed, for what purpose, and by which means. This also includes information about the ways the data are protected. Data subjects should also be informed about third parties with whom information is shared.

Possible design patterns in this category are Transparency, and the Platform for Privacy Preferences (P3P)[21] – although the latter also fits the *control* principle. Systems for data breach notifications are also a design pattern in this category.

3.2.6. *Control*

The *control* principle states: *Data subjects should have agency over the processing of their personal data*. The *control* principle is in fact an important counterpart to the *inform* principle. Without reasonable means of controlling the use of one's personal data, there is little use in informing a data subject about the fact that personal data is collected. Data protection legislation often gives the data subject the right to view, update and even ask the deletion of personal data collected about him.[22] This principle underlines this fact, and design patterns in this class will give users the tools to exert their data protection rights.

Control also governs the means by which users can decide whether to use a certain system, and the way they control what kind of data is processed about them. In the context of social networks, for example, the ease with which the user can update their privacy settings through the user interface determines the level of control to a large extent. So user interaction design is an important factor as well.

Design patterns are: informed consent and certain user interaction design patterns.

3.2.7. *Enforce*

The seventh principle, *enforce*, states: *A privacy policy compatible with legal requirements should be in place and should be enforced*. The enforce principle ensures that the system is compatible with data protection legislation, both at the time when the system is developed and when the system is in operation. This requires periodic evaluations and where necessary updates in case changes occur in legislation. By specifying a privacy policy, and setting up the appropriate governance structures to enforce that policy, proper embedding of the IT system within the organisation is established.

Design patterns are: access control, and privacy rights management – a form of Digital Rights Management (DRM), but then applied to privacy that is discussed extensively further below.

3.2.8. *Demonstrate*

The final strategy, demonstrate, requires a data controller to *be able to demonstrate compliance with the privacy policy and any applicable legal requirements*. This strategy goes one step further than the enforce strategy in that it requires the data controller to prove that it is in control. This is explicitly required in the new draft EU privacy regulation. In particular this requires the data controller to be able to show how the privacy policy is effectively implemented within the IT system. In case of complaints or problems, she should immediately be able to determine the extent of any possible privacy breaches, for example.

Design patterns that implement this strategy are, for example, privacy management systems,[23] and the use of logging and auditing.

[20] See for instance, artt. 10 and 11 of Directive 95/46/EC on the protection of individuals with regard to the processing of personal data and on the free movement of such data OJ L281, 23/11/1995.

[21] http://www.w3.org/P3P/.

[22] See, for instance, art. 12 of Directive 95/46/EC on the protection of individuals with regard to the processing of personal data and on the free movement of such data OJ L281, 23/11/1995.

[23] Casassa Mont, Marco and Pearson, Siani (2005). An adaptive privacy management system for data repositories. In Sokratis K. Katsikas, Javier Lopez, and Günther Pernul, editors, Trust, Privacy and Security in Digital Business: Second International Conference TrustBus 2005, Copenhagen, Denmark, August 2226, 2005, Proceedings, LNCS 3592, dordrecht etc: Springer, pp. 236–245.

3.3. *Revocable privacy*

As mentioned in the introduction on each of the design strategies above, there are many basic privacy-enhancing design patterns that can be used to implement the design strategies. Which design pattern is appropriate depends on the goals to be achieved (e.g., limiting collection of data or limiting use of data), but also the context in which it will be applied. For instance, pseudonyms can be used to hide the relationship between a real natural person and some personal data. Replacing real names by nicknames or arbitrary numbers can serve this purpose, as long as the relationship between the pseudonym and the real person remains a secret for those for whom the identity of the natural person needs to be hidden. Note however that in many cases the use of pseudonyms in for example online profiling is pretty much irrelevant: the online profile can be applied whenever the same user is recognised, without ever knowing his real identity. Anonymous credentials, another design pattern, are a form of anonymous attribute certificates. Similar to attribute certificates, such credentials express a property about a subject (like 'the bearer of this credential is over 18 years old') that is signed by an authority that can verify the validity of the claim. Traditional attribute certificates are not anonymous, because the same certificate is presented whenever one wants to convince someone about the validity of a claim. These credentials, obviously, are only useful in contexts where validated claims play a role. In the context of open-source intelligence, where for instance Facebook is mined to find indicators for certain suspect behaviour or attitudes, anonymous credentials are not useful.

In the remainder of the paper we will concentrate on two categories of privacy-enhancing techniques that seem useful in the context of open-source intelligence. The first (revocable privacy) highlights that privacy can be a default, to be lifted when required. The second (data dandling policies and policy enforcement, Section 3.4) highlights that the use of personal data can potentially be regulated through technology.

Revocable privacy[24] builds on the insight that legal or regulatory requirements for data minimisation do not always work, and the solution must be found in limiting possibilities at the outset, through PETs, in the architecture and design of the system (Galindo and Hoepman, 2011). In essence, the idea of revocable privacy is to design systems in such a way that no personal data are available, unless (pre-established) conditions are met that necessitate lawful access to the identifying information of a specific individual. Only in that case, the personal details and when and how the predefined conditions have been met, are revealed. The data are only revealed to authorised parties, and the guarantees are technical rather than legal in nature. An early example of revocable privacy is the system proposed by David Chaum and others for a scheme for off-line digital cash[25]; people could use this digital cash anonymously, unless someone double spends a coin, in which case the identity of the owner of the coin would be revealed.

We define revocable privacy as follows:

> *a system implements revocable privacy if the architecture of the system guarantees that personal data is revealed only if a predefined condition has been met.*

We distinguish two variants of revocable privacy.

a) *Spread responsibility*

One or more trusted third parties verify whether all conditions for releasing personal data have been met, and grant access (or release the data) if this is the case.

b) *Self-enforcing architecture*

The rules to release data are hard-coded into the architecture. If the condition is met, the data are released automatically. If the condition has not been met, no information can be obtained at all.

Many of the techniques currently in use for revocable privacy are based on the use of trusted third parties. By spreading the power over several such parties (using secret sharing techniques or similar), one can mitigate the likelihood of corruption or subversion. Such systems are in essence procedure-based, and they can in principle be thwarted by changing the procedures and, for example, replacing the trusted parties. Although the self-enforcing approach to revocable privacy is to be preferred from that perspective, it requires hard-coding the condition(s) under which personal data should be released, and it depends on the type of rule whether this can be effectively translated into software code.[26] If that is not feasible, the approach of spread responsibility is a good, if sub-optimal, alternative.

Whether revocable privacy is a feasible approach for open-source intelligence will depend on several factors: the nature of the investigation (e.g., whether it focuses on individuals or on objects or broader trends), the relevance for the investigation of mapping networks of individuals, the precision with which the identities of relevant individuals can be recognised by the system, the patterns that one wants to detect, and the stage(s) of the investigation in which recognisable individuals or connections between individuals need to be analysed. This requires an in-depth analysis for specific OSINT settings. We will provide some reflections on the possibilities of applying revocable privacy in an OSINT context in Section 4.

3.4. *Enterprise privacy policies and technologies for legal compliance*

Implementing the privacy-by-design strategies as outlined above is desirable for any system. Within the European context there is more guidance as to what requirements an IT system that processes personal data must meet. These requirements are embedded in the data protection regulation, in

[24] Hoepman. J.-H. (2009), 'Revocable Privacy'. 5 *ENISA Quarterly Review* (2), pp. 16–17.

[25] Chaum, David, Amos Fiat, and Moni Naor (1988). 'Untraceable electronic cash'. In Shafi Goldwasser (ed.), *CRYPTO*, LNCS 403, Dordrecht etc: Springer, pp. 319–327.

[26] Koops, B.J. (2011), 'The (In)flexibility of Techno-Regulation and the Case of Purpose-Binding', 5 *Legisprudence* (2), pp. 171–194.

particular the Data Protection Directive (hereafter: DPD) and the associated Framework Decision.[27] The entire system (encompassing both IT and organisational structures and procedures) has to comply with the data protection legislation. The threshold for data to be qualified as personal data is rather low: any information relating to an identifiable or identified natural person is personal data. Hence, names, unique identification numbers, but also combinations of data that allow an individual to be singled out in a set of individuals, and pictures with recognisable individuals qualify as personal data. Research of OSINT practices indicates that these practices process personal data and hence need to comply with the DPD (or, in case of police and judicial data processing, with the associated Framework Decision).[28]

Part of the legal compliance can be supported by technology and different parts of the regulation afford or require different technologies. For our current purposes, the DPD can be divided into five relevant blocks. The first block relates to the applicability of the DPD. This includes determining whether personal data is processed, who the data processor is, whether this data controller falls within the ambit of the DPD, whether one of the exceptions (such as the household exemption) applies. For developing an IT system this test only has to be done once, and for OSINT platforms such as VIRTUOSO we can assume that the DPD applies. End-users will have to make separate tests whether their intended use of the platform and additional tools falls within the DPD ambit, or whether they fall under the exemption of, for example, public security or defence. The second block pertains to the legitimacy of data collection. Again, in principle this only has to be verified once provided the way in which 'the system' collects personal data is stable. The third block relates to the further processing of personal data. Here the assumption is that data is being stored and needs to be retrieved in order to be further processed (checked, modified, deleted, etc). Whether or not such activities are permitted on certain data needs to be determined during runtime and depends on factors such as the purposes for which the data were collected in the first place, who or what processes the data, what kind of data is being processed, etc. A fourth set of requirements deals with information obligations on the data controller. For instance, the data controller has a duty to notify the data protection authorities of the intended data processing and needs to inform the data subjects whose data they will be processing. A fifth block pertains to requirements regarding data quality and security. Data quality includes requirements such as that only data necessary for a certain purpose may be processed, these data need to be accurate and not excessive, and may only be retained for as long as necessary for the purpose for which they were collected. Data security means that adequate and cost-effective measures need to be taken to guarantee the integrity, availability, and confidentiality of the data in the system. The rest of the provisions in the DPD have less practical relevance for OSINT platforms.

Compliance with these five blocks can in principle be supported by technology. For instance, one can envision expert systems or wizards that guide in-company Privacy Officers (or those responsible for legal compliance within relevant OSINT settings) through the first block of requirements to establish whether or not the implementation of a particular OSINT platform or application falls within the scope of the DPD. But since this is a one-time assessment that can also be done on the basis of written material, this does not seem to be a relevant technical support.

More relevant is monitoring the collection and processing of personal data in the runtime environment. The collection and further processing of personal data is regulated by the DPD and by policies established by the data controller. An example of the former is that sensitive personal data (such as information about race, disease, or political belief) may only be processed with prior explicit consent of the data subject. More important are the processing requirements as established by the data controller. Article 6 DPD specifies that data controllers may only process personal data for specified, explicit and legitimate purposes. As processing includes collection as well as treating and storing data, this places a requirement on data controllers to make explicit what data they will process in all stages of their activities and for which purposes they will do so. Once these purposes are specified, the data controller is bound to them. In other words, data controllers need to specify Enterprise Privacy Policies. These policies are based on the enterprise's goals and need to be compatible with the DPD.

Enterprise Privacy Policies are based on the enterprise's goals and operation. They should be distinguished from privacy statements. Privacy policies are 'internally focused tools describing how an organisation intends to achieve the [data protection] principles set out [in personal data protection legislation] and a clear means to provide for accountability'.[29] By contrast, privacy statements are 'externally facing tools supporting objectives of transparency, [which] would alert individuals at an appropriate time and context as to how their personal data is being used'.[30] In other words, privacy policies are intended to provide 'a set of rules for employees, members and member organisations to follow (…) and provide important guidance about correct procedure and behaviour based on a version of information privacy principles'.[31] Privacy policies are internal measures of companies and other organisations to ensure data protection compliance of their organisational processes. Privacy statements on the other

[27] Directive 95/46/EC on the protection of individuals with regard to the processing of personal data and on the free movement of such data OJ L281, 23/11/1995; Framework Decision 2008/977/JHA regarding the processing of personal data in the framework of police and judicial cooperation in criminal matters, OJ L350, 30/12/2008. For brevity's sake, we will focus on the general DPD in this paper.

[28] Koops, B.J., C. Cuijpers & M. Schellekens (2011), *D3.2: Analysis of the Legal and Ethical Framework in Open Source Intelligence*, VIRTUOSO deliverable, 1 December 2011, http://www.virtuoso.eu/VIRTUOSO/servlet/document.fileView/Virtuoso-D3%202-Legal%20and%20ethical%20constraints-final-2011-12-21.pdf; see also the article by C. Cuijpers in this issue.

[29] Robinson, N. et al. (2009), *Review of the European Data Protection Directive*, Santa Monica, CA: RAND.

[30] Bennett, Colin J. and Charles D. Raab (2007), *The Governance of Privacy. Policy instruments in a global perspective*, Cambridge: The MIT Press.

[31] *Ibid*.

hand are external, often rather brief communications of organisational privacy policies that do not comprehensibly reflect the complexities of organisational personal data processing procedures.

Enterprise Privacy Policies (EPPs) consist of data handling and data access policies. They specify the conditions under which certain personal data will be collected/processed and for which purposes. These policies can hence be used to monitor the data-collection process and can be enforced during further processing; they resemble (DRM) systems used to enforce copyright compliance, but in this case applied to privacy compliance. If all data processing within a system is described in data access and data handling policies and the actual personal data contain the appropriate metadata, a policy engine can then enforce the policies during runtime. This can ensure that only authorised data requests are honoured for the right purposes.

Compliance can now be specified in more detail:

a) The system should apply the data access and data handling policies correctly to requests for personal data by the back-end system(s). The IT system should correctly enforce the enterprise privacy policies.
b) The system should correctly enforce also those legal obligations that have no counterpart in EPPs. In other words, ideally, the system also will have general data protection requirements built in.
c) The norms (EPPs) to be transposed into executable form need to be compliant with prevailing legal regulation.
d) The norms (EPPs) as formulated by the Privacy Officer should be correctly transposed into executable form. In other words, the runtime system should operate on the correct norms (syntax).
e) The norms as formulated by the Privacy Officer should represent what they intended to represent (semantics).

In terms of the design principles outlined in Section 3.2, these requirements can be implemented as part of the enforce principle. Technical tools that help enforce compliance of data handling and data access policies and legal obligations are still at a relatively early stage of development. In the past years, several systems have been developed for technology-assisted legal compliance. These include P3P, EPAL,[32] XACML-based approaches,[33] and the work done in the EU FP7 ENDORSE project which aims to provide technical tools for enforcing privacy policies and legal compliance.[34]

[32] IBM (2004), *The enterprise privacy authorization language (epal)*, http://www.zurich.ibm.com/security/enterprise-privacy/epal/.

[33] See for a brief discussion Koops (n 26) pp. 183–187; Franz-Stefan Preiss (2012), Minimizing Information Disclosure in Authentication Transactions with Attribute-Based Credentials, Leuven. See also Section 4.2.1.

[34] Olislaegers, S. (2012), 'Early Lessons Learned in the ENDORSE Project: Legal Challenges and Possibilities in Developing Data Protection Compliance Software', in Camenisch, Jan, Crispo, Bruno, Fischer-Hübner, Simone, Leenes, Ronald, Russello, Giovanni (eds.), Privacy and Identity Management for Life, IFIP Advances in Information and Communication Technology Volume 375, Dordrecht etc: Springer, pp. 73–87.

4. Application of privacy by design in OSINT

4.1. *Legal requirements that are amenable to privacy by design*

Zooming in on privacy and data protection requirements, the following requirements have been identified from data protection law[35] as having the most potential for a techno-regulation approach.[36]

1. **Purpose specification**: the purpose(s) for the collection and use of personal data must be specified, with as much precision as possible (art. 6 b Directive 95/46/EC).
2. **Legal basis or legitimate ground**: Is there a legal ground (art. 6 a and 7 Directive 95/46/EC) that legitimates the processing (including the collection)? I.e., is there (informed and explicit) consent of data subjects, a legal obligation, or a substantial interest that outweighs the privacy interest of the data subjects?
3. **Collection and use limitation/data minimisation**: the collection and use of the personal data must be restricted to what is necessary for fulfilling the purpose(s) defined according to requirement 2 (art 6 c Directive 95/46/EC).
   a. The principle of **subsidiarity** applies: the purposes cannot be fulfilled by less invasive means than the proposed collection and use of the personal data.
   b. The principle of **proportionality** applies: the collection and processing must be not excessive in relation to the defined purpose(s).
   c. '**Select** before you collect'.
   d. The principles of subsidiarity and proportionality likewise apply to transferring data to third parties: this can only be done if necessary for the specified purpose(s) and if proportionate to achieving these.
   e. The **default** settings should be set in such a way that privacy of data subjects is maximally protected, so that no action by data subjects is required to enhance the privacy in the system.
   f. The **period** of time for which the personal data are kept, must be restricted to a minimum. As soon as data has served to fulfil the purpose(s) for which it was collected, it should be **destroyed** (art. 6 e Directive 95/46/EC).
   g. The re-use of the data for purposes **incompatible** with the original purpose is not permitted (art. 6 e Directive 95/46/EC). (This is only permissible for historical, statistical or scientific purposes, which are not likely to apply to OSINT systems used for security purposes).
   h. If the product can be used for different purposes or be run in a **multi-user environment** (i.e. virtually connected systems, such as data warehouses, cloud computing, digital identifiers), the system design should be such that data and processes serving different tasks or purposes can be segregated from each other in a secure way.

[35] We base ourselves here on the European Data Protection Directive 95/46/EC.

[36] Koops et al. (n 1).

4. **Data quality**: the personal data collected and used by the product must be (art. 6 d Directive 95/46/EC):
   a. up-to-date;
   b. reasonable measures must be taken to ensure that data which are inaccurate or incomplete (in light of the purpose(s) for which they were collected) are **erased or rectified**;
5. **Rights of data subjects**: data subjects must be allowed to exert their information rights (art. 12 Directive 95/46/EC):
   a. availability of contact information: this must be provided generally, so that data subjects know whom they can contact to exercise their rights;
   b. right to information: persons must be able to ask and receive information whether and if so, for what purposes their data are processed;
   c. right of correction and removal: persons must be able to ask and be ensured that incorrect data are corrected or that data are destroyed.
   d. the possibilities regarding these rights, including the right to information and objection, should be supported by technological means.
6. **Security safeguard**s (art. 17 Directive 95/46/EC): appropriate measures must be used by the OSINT system and its end-users to safeguard the security of their systems and the processed data, prevent unauthorised access to data, secure use and disposal, security awareness and training, etc.
   a. It is necessary to design and secure the systems in a way that only authorised entities have access to personal data.
   b. If unlawful use of the platform or certain components is to be expected, the framework should be designed in such a way that it prevents (preferably by making it impossible or extremely hard) this type of unlawful use.
   c. If the data resulting from OSINT are to be used as evidence in criminal proceedings, then requirements and procedures for collecting reliable digital evidence will apply, otherwise the data will be inadmissible or considered unreliable in court.
   d. Notification protocols must be developed for the event of a privacy breach.
   e. Redress protocols must be developed for the event of a privacy breach.

Whether and to what extent these requirements can effectively be implemented in the design of an OSINT system or platform is highly context-dependent. It is beyond the scope of this paper to provide an in-depth analysis of all requirements in relation to a particular instantiation of an OSINT platform. Instead, we will discuss three requirements in relation to a fairly generic level of OSINT platforms, to illustrate to what extent a privacy-by-design approach may be feasible using the privacy design strategies and how policy-enforcement language or revocable privacy could function to implement the principles. The requirements chosen for further elaboration in this article cover the most important control in the OSINT realm: making sure that data is only used for the right purposes, that data is deleted whenever possible and that correct data is used.

4.2. *Illustration of sample requirements*

4.2.1. *Purpose specification and use limitation*

At the highest level of abstraction, these requirements can be fulfilled using the privacy design strategies of *minimise*, *separate*, and *enforce*. The design pattern that can best be used for meeting these requirements is policy enforcement.

We illustrate this by focussing on the policy-enforcement language XACML (eXtensible Access Control Markup Language). XACML is an XML-based language to express and interchange access control policies expressed as authorisation policies against objects that are themselves identified in XML.[37] It also specifies the syntax and format of obligations and defines an architecture for the evaluation of policies (by a Policy-Enforcement Point) and a communication protocol for the message exchange. XACML has been used, for instance in the EU FP7 PrimeLife project[38] to implement access control based on privacy policies.

An XACML policy document consists of a set of Access Control Rules. A Rule consists of one or more Conditions and a Rule effect, which is either Permit or Deny access. If the Rule's conditions are met, the Rule's effect obtains (Permit or Deny). If the conditions are not satisfied, NA (not applicable) is returned.

Such rules can be used to determine under which circumstances an actor or process may gain access to a particular (class of) resource(s). They can be associated to particular data (such as a multimedia file) specifying the conditions for access (such as actor is 'mandated law enforcement agent' and purpose is 'ongoing concrete investigation'). Alternatively general rules can be matched against properties of the requesting actor (mandated official) or process and metadata associated to resources that, for instance specify purposes for which the data may be used.

When a user or process wants to gain access to a resource (for instance certain personal data) A Policy-Enforcement Point (PEP) will check whether the properties of the requester meets the conditions specified in the Access Control Rules associated with the requested resource (case 1), or whether there is an access control rule that matches properties of requestor and conditions specified in the metadata. If there is a match the PEP will allow access, else refuse access.

Although it was not developed for implementing legal compliance, it can be used as such because it is possible to frame part of the data protection legal framework in terms of access control rules. For instance, the data access and data handling rules specified in an Enterprise Privacy Policy (see Section 3.4) are amenable to implementation as access control rules. Such rules should be closely linked to the purposes specified for individual investigations using an OSINT platform. Depending on the type and scope of investigation, fewer or more people can be granted access rights to the data

[37] Ardagna, Claudio A., Sabrina De Capitani Di Vimercati, Gregory Neven, Stefano Paraboschi, Eros Pedrini, Franz-Stefan Preiss, Pierangela Samarati, and Mario Verdicchio (2011), 'Advances in Access Control Policies', in: Camenisch, J., S. Fischer-Hübner, K. Rannenberg (eds), Privacy and Identity Management for Life, Heidelberg, Dordrecht: Springer.

[38] http://primelife.ercim.eu/.

collected by use of the platform. To prevent unlawful secondary use of data, the data from one investigation should by default not be accessible for analysts working on another investigation, unless the purpose of the second investigation is compatible with that of the former. By connecting access rights to prior specified purposes, the uses of collected data can be limited to those that are necessary in light of the primary purpose, and possibly also compatible secondary purposes, thus safeguarding the principle of use limitation.

Although XACML thus provides a useful tool for policy enforcement, employing it to achieve real purpose specification and use limitation will not be straightforward in practice. Particularly if the purposes of data processing are non-standard and/or rather general or vague, it will be difficult to bind strict access control rules to the data.[39] Nevertheless, the potential of XACML is interesting, so that it seems a useful suggestion for OSINT platform developers to consider extending the platform with an XACML – or equivalent policy-enforcement language – feature.

4.2.2. *Retention while necessary, deletion when possible*

At the highest level of abstraction, this requirement should be fulfilled using a combination of the following privacy design strategies: *hide* and *enforce*. With the exception of a privacy management system, none of the design patterns mentioned above really fits this requirement exactly. A recent methodology that comes close to the idea of 'retention while necessary, deletion when possible' is revocable privacy (see Section 3.3).

In the revocable privacy approach, identifiers of individuals are encrypted (or replaced by a pseudonym with the mapping of identifiers and pseudonyms being encrypted); the data relating to these identifiers can then only be connected to the individuals by decryption, which is only possible if a predefined rule is triggered. The approach therefore consists of two elements: encrypting identifiers by default, and decrypting identifiers when allowed.

As to the first element, in an OSINT context (or any intelligence context in which data are automatically processed), it may seem counterintuitive to apply this, since the business of intelligence is, after all, to find as much and as precise information as possible and relevant in relation to a certain issue. However, in some contexts, intelligence does not focus on specific individuals, but rather on objects (such as a nuclear installation) or phenomena (such as arms trafficking). Moreover, in cases that do target specific individuals, it is not always necessary to know details of other individuals who show up as secondary findings in the investigation.

More problematic, however, is the technical issue of connecting data from various sources in relation to individuals; even if the investigation does not necessarily need to have personally identifying information about individuals, it will be important to match information from different sources that relate to the same persons. Open-source information will often contain unchecked information in which spelling variants and typos may occur in people's names or other identifying information. OSINT systems need to be designed in such a way that analysts will be able to determine which information from different sources relate to the same person. It cannot be determined within the scope of this report whether this presents an insurmountable obstacle to applying revocable privacy in OSINT operations. Perhaps techniques similar to homomorphic encryption, which allow searching in encrypted data or making computations on encrypted data without revealing the data themselves,[40] could offer possibilities for relating encrypted data from different open sources to each other. No techniques exist (yet) for connecting encrypted identifiers with slight spelling variations; this would be a daunting task to develop but not necessarily impossible, so this is a relevant topic for further research. In the longer term, this might provide opportunities for revocable privacy in the data-collection stages of OSINT.

For the time being, while such an approach is not (yet) technically infeasible for matching data from different sources, revocable privacy can still be applied *after* analysts have checked the data but *before* data are stored in databases or further processed for decision-making. Encrypting identifiers of people who are determined by the analysts to be not directly relevant for further investigation but whose data have to be kept for possible future stages of the investigation, could be a relevant way of protecting the personal data of these individuals. But also for people who *are* directly relevant for the investigation, it need not always be necessary to identify them by their real names in reports or databases. For certain decisions in the course of the investigation, pseudonyms may work equally well, and not all people with authority to access reports or databases need to know the identifiers of target persons. It is in this respect that the concept of revocable privacy may really have added value for OSINT operations: only when there is a necessity of someone in the end-user organisation knowing real identifiers of data subjects – and the situations in which the officials for which this is necessary can be described in authorisation protocols – will the identifiers be decrypted. This provides an extra layer of privacy protection as well as security, not only against data leaks and against internal misuse, but also against situations in which individuals are wrongly targeted, for instance in cases of identity theft or false positives in profiling.

Supposing that some form of revocable privacy is possible, either automatically when data are collected and automatically processed, or after data have been checked by analysts, the second element is that personal information will be made accessible when a certain predefined rule is triggered. In the context of OSINT systems, the self-enforcing variant of revocable privacy (see Section 3.3) is unrealistic because of the complexity of the rules under which personal information should or should not be available. It is however possible to make use of Trusted Third Parties and apply the spread responsibility approach.

Recall that in the spread responsibility approach (see Section 3.3) the decision to allow access to a specific item containing personal data is spread over several parties. That is to say that access to the data item is only possible if each of the pre-assigned parties has given consent for the specific access request. More complex arrangements are also possible. For example, it could be made mandatory to get consent from a

[39] See Koops (n 26) for an analysis.

[40] See http://en.wikipedia.org/wiki/Homomorphic_encryption.

certain sub-group of people from a larger, pre-determined, group, for example of privacy officers, auditors and board members.

From a technical perspective, the spread responsibility approach can be implemented in several ways. A straightforward approach is to use a method similar to that of the previous section to implement purpose specification and use limitation. If a user or process wants to gain access to a resource (for instance certain personal data), a request is forwarded to the Policy-Enforcement Point (PEP) which acts as a gateway to the resource. The corresponding Policy Decision Point collects the access conditions and collects the explicit consent from the required set of people authorised to give permission to de-anonymise personal data.

A more secure implementation of the spread responsibility approach can be based on secret sharing techniques.[41] In this approach, data items are all encrypted, making them inaccessible by default. The corresponding decryption key is shared across the group of officers that are entrusted with making access decisions. Each officer is given a share of the decryption key. By itself, this share is useless (and does not give any information about the decryption key). However, by combining enough shares (above a predefined threshold) of the decryption key, the decryption key can be recovered and the corresponding data item can then be decrypted.

4.2.3. *Personal data must be up-to-date*

At the highest level of abstraction, this requirement should be fulfilled using a combination of the following privacy design strategies: *enforce* and *control*. Privacy management systems, together with the use of appropriate metadata, can be used to enforce procedures that check the integrity and the 'age' of the personal data that is maintained by a system. When discrepancies or over-aged items are detected, they can be marked as such, they can be deleted, or an action can be triggered to update the information (e.g. by automatically checking the original open sources from which the data were retrieved).

By giving end-users, who also have an interest in using the most up-to-date information, user-friendly tools to manage the personal data they collect and store in various investigations, personal data can be better kept up-to-date. Instead of end-users managing personal data centrally, the storage could be distributed using some kind of identity management system[42] or one of the digital locker products that are now emerging (Singly, Qiy).[43]

5. Conclusion

Technology enables privacy infringements but it can also enable privacy safeguards. A techno-regulation approach as discussed in this paper provides a relevant means for diminishing the potential privacy impact of OSINT systems, such as the VIRTUOSO platform. Adopting this approach is more than merely ticking away a check list of requirements.[44] Privacy by design requires an attitude in product, system and architecture design that thinks in terms of privacy protection and the interests of individual persons, and that looks for win–win approaches to address privacy concerns along with other system requirements such as usability, security, and cost-effectiveness.

Various data protection requirements should be considered as important candidates for being protected through privacy by design: purpose specification; collection and use limitation (e.g., through default settings, enforceable deletion after the necessary retention period, purpose-binding); data quality; not processing sensitive data (including images) unless absolutely necessary and with additional safeguards; and respecting access restrictions to data.

However, it must be acknowledged that end-users may have contrary requirements. For example, many end-users will want to collect visual information, such as photos and videos, and end-users in government security contexts will generally also want to collect data from websites that indicate (by means of e.g. a robot.txt file or metadata) that they do not want to be searched by crawlers.[45] Legal requirements are not absolute and are open to interpretation.[46] Therefore, it seems inappropriate to 'hard-wire' these requirements in an OSINT platform; rather, it could be left to end-users to decide for themselves, and at their own risk of liability for privacy and intellectual-property infringements, to what extent they want to use plug-ins for retrieving visual information and overriding crawler-access stipulations. Still, from the perspective of responsible design, it is useful for OSINT platform providers to include at least certain warnings for end-users about these issues, for example pop-up screens with basic information about the legal risks of collecting photos and videos and overriding crawler-access restrictions.

Requirements relating to purpose specification, use limitation and data quality may be more compatible with end-user requirements, although there will be widely diverging views on the period for which data should be retained and on the issue of secondary processing of collected data.[47] This underlines the usefulness of a policy-enforcement approach, which enables end-users to stipulate their own requirements in an Enterprise Privacy Policy, indicating for which purposes the OSINT platform and plug-ins can be used, how long data should be stored, and who can access collected data for which purposes and under which conditions.

By incorporating a policy-enforcement language, such as XACML (eXtensible Access Control Markup Language), in an OSINT platform, end-users can express access control policies linked to the purposes specified for individual investigations using the platform. Depending on the type and scope of

[41] Shamir, Adi (1979), 'How to share a secret', 22 *Communications of the ACM* (11), pp. 612–613.

[42] Cf. Alpár, Gergely, Jaap-Henk Hoepman & Johanneke Siljee, 'The Identity Crisis. Security, Privacy and Usability Issues in Identity Management'. *Journal of Information System Security*, 2013. (to appear 2013).

[43] See http://blog.singly.com/tag/locker-project/, respectively https://www.qiy.nl/nl.

[44] Gürses, S., C. Troncoso and C. Diaz (2011), 'Engineering Privacy by Design', *Proceedings CPDP 2011*.

[45] Cf. the article by Maurice Schellekens in this issue.

[46] See Koops, Cuijpers and Schellekens (n 27), discussing to what extent photographs contain sensitive personal data and discussing the legal status of robot.txt files.

[47] See Koops et al. (n 1) for an overview of end-user requirements for OSINT systems.

investigation, fewer or more people can be granted access rights to the data collected by use of the platform. To prevent unlawful secondary use of data, the data from one investigation should by default not be accessible for analysts working on another investigation, unless the purpose of the second investigation is compatible with that of the former.

Although XACML thus provides a useful tool for policy enforcement, employing it to achieve real purpose specification and use limitation will not be straightforward in practice. Particularly if the purposes of data processing are non-standard and/or rather general or vague, as may be the case with open-source intelligence investigations, it will be difficult to bind strict access control rules to the data. Nevertheless, the potential of XACML is very relevant, so that OSINT platform developers must consider including an XACML – or equivalent policy-enforcement language – feature in their platform.

The policy-enforcement system could be combined by end-users with a revocable privacy approach, which enforces the requirement of data minimisation. In this approach, as soon as data collected from open sources are recognised with sufficient precision as containing personally identifiable information (which in the current state-of-the-art will be after analysts have checked and linked relevant data), the identifiers (such as people's names, addresses, email addresses, administration numbers) are pseudonymised (using encryption). The personal data will only be made accessible when a certain predefined rule is triggered, based on authorisation and purpose limitation. The decision to allow access to a specific item containing personal data can be spread over several parties, for example the organisation's Data Protection Officer and the Head of the Intelligence Unit responsible for investigations. Both parties would have to agree that releasing the key to de-pseudonymise OSINT-collected data is relevant for the investigation and is compatible with the purposes for which the data were originally collected. More complex arrangements are also possible, for example by making it mandatory to get consent from a certain sub-group of people from a larger, pre-determined, group, for example of privacy officers, auditors and board members.

To meet the requirement that personal data must be up-to-date, also privacy management systems can be used, in combination with the use of appropriate metadata, to enforce procedures that check the integrity and the 'age' of personal data stored by the system. When discrepancies or over-aged items are detected, they can be marked as such or deleted, or action can be triggered to update the information (e.g. by periodically checking the original open sources from which the data were retrieved).

Ultimately, it will be end-users who determine to what extent privacy by design will really be embraced when they adopt an OSINT platform, extend it with plug-ins, and fine-tune it to their needs. But OSINT platform providers at least have a responsibility to make sure that end-users are enabled to use privacy by design, by allowing functionalities such as revocable privacy and a policy-enforcement language to be part of their platform. In that way, the developers send a clear signal to end-users that stresses the vitality of data protection in today's world.

Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007–2013) under grant agreement n° FP7-SEC-GA-2009-242352. The authors thank Gabriela Bodea, Gérard Dupont and Edward Hobbs for providing valuable input for this research.

Prof. Dr. Bert-Jaap Koops (*e.j.koops@tilburguniversity.edu*) *Professor of Regulation and Technology, TILT – Tilburg Institute for Law, Technology, and Society, Tilburg University, the Netherlands.*

Jaap-Henk Hoepman *Senior Scientist in computer security, privacy and identity management TNO, (the Dutch Organisation for Applied Scientific Research) and; Institute for Computing and Information Sciences, Radboud University Nijmegen, The Netherlands.*

Prof. Dr. Ronald Leenes *Professor in Regulation by Technology, TILT – Tilburg Institute for Law, Technology, and Society, Tilburg University, the Netherlands.*

REFERENCES

Alpár Gergely, Jaap-Henk Hoepman, Siljee Johanneke. The identity crisis. Security, privacy and usability issues in identity management. J Inf Syst Secur 2013 in press.

Ardagna Claudio A, De Capitani Di Vimercati Sabrina, Neven Gregory, Paraboschi Stefano, Pedrini Eros, Preiss Franz-Stefan, et al. Advances in access control policies. In: Camenisch J, Fischer-Hübner S, Rannenberg K, editors. Privacy and identity management for life. Heidelberg, Dordrecht: Springer; 2011.

Bennett Colin J, Raab Charles D. The governance of privacy. Policy instruments in a global perspective. Cambridge: The MIT Press; 2007.

Black J. What is regulatory innovation? In: Black J, Lodge M, Thatcher M, editors. Regulatory innovation. Edward Elgar; 2005. p. 103–46.

Brownsword R. What the world needs now: techno-regulation, human rights and human dignity. In: Brownsword R, editor. Global governance and the quest for justice. Human rights, vol. 4. Hart; 2004. p. 203–34.

Buschmann F, Meunier R, Rohnert H, Sommerlad P, Stal M. Pattern-oriented software architecture: a system of patterns. John Wiley and Sons; 1996.

Camenisch R, Fischer-Hübner S, Rannenberg K, editors. Privacy and identity management for life. Dordrecht|Heidelberg: Springer; 2011.

Casassa Mont Marco, Pearson Siani. An adaptive privacy management system for data repositories. In: Katsikas Sokratis K, Lopez Javier, Pernul Günther, editors. Trust, privacy and security in digital business: second international conference, TrustBus 2005. Springer; 2005. p. 236–45. Copenhagen, Denmark, August 22–26, 2005, Proceedings, LNCS 3592.

Chaum David, Fiat Amos, Naor Moni. Untraceable electronic cash. In: Goldwasser Shafi, editor. CRYPTO, LNCS 403. Springer; 1988. p. 319–27.

Galindo D, Hoepman J-H. Non-interactive distributed encryption: a new primitive for revocable privacy. In: Workshop on

Privacy in the Electronic Society (WPES) 2011. p. 81–92. Chicago, IL, USA, October 17 2011.
Gürses S, Troncoso C, Diaz C. Engineering privacy by design. In: Proceedings CPDP 2011 2011.
Hoepman J-H. Revocable privacy. ENISA Q Rev 2009;5(2):16–7.
Hoepman J-H. Privacy design strategies eprint, arXiv:1210.6621; October 2012.
Jacobs B. Select before you collect. Ars Aequi 2005;54:1006–9.
Koops BJ, Cuijpers C, Schellekens M. D3.2: analysis of the legal and ethical framework in open source intelligence, VIRTUOSO deliverable. http://www.virtuoso.eu/VIRTUOSO/servlet/document.fileView/Virtuoso-D3%202-Legal%20and%20ethical%20constraints-final-2011-12-21.pdf; 1 December 2011.
Koops BJ. The (in)flexibility of techno-regulation and the case of purpose-binding. Legisprudence 2011;5(2):171–94.
Leenes RE. Harde Lessen: Apologie van technologie als reguleringsinstrument. Tilburg: Universiteit van Tilburg; 2010. p. 21.
Lessig L. Code and other laws of cyberspace. New York: Basic Books; 1999.
Olislaegers S. Early lessons learned in the ENDORSE project: legal challenges and possibilities in developing data protection compliance software. In: Camenisch Jan, Crispo Bruno, Fischer-Hübner Simone, Leenes Ronald, Russello Giovanni, editors. Privacy and identity management for life. IFIP advances in information and communication technology, vol. 375; 2012. p. 73–87.
Robinson N, Graux H, Botterman M, Valeri L. Review of the European data protection directive. Santa Monica, CA: RAND; 2009.
Shamir Adi. How to share a secret. Commun of the ACM 1979;22(11):612–3.