# Sticky Policies: A Survey

Daniele Miorandi, Alessandra Rizzardi, *IEEE member,* Sabrina Sicari, *IEEE member,*
and Alberto Coen-Porisini

**Abstract**—In the digital age, where the Internet connects things across the globe and individuals are constantly online, data security and privacy are becoming key drivers (and barriers) of change for adoption of innovative solutions. Traditional approaches, whereby communication links are secured by means of encryption, and access control is run in a static way by a centralised authority, are showing their limits when applied to massive-scale, interconnected and distributed systems. Regulations, while still fragmented, are moving to adapt to changes in technology and society, with the aim to protect confidential information by governments, businesses, and individual citizens. In this landscape, proper mechanisms should be defined to allow a strict control over the data life-cycle and to guarantee the privacy and the application of specific regulations on personal information's disclosure, usage and access. Sticky policies represent one approach to improve owners' control over their data. In such an approach, machine-readable policies are attached to data. They are called 'sticky' in that they travel together with data, as data travels across multiple administrative domains. In this article we survey the state-of-the-art in sticky policies, discussing limitations, open issues, applications and research challenges, with a specific focus on their applicability to Internet of Things, cloud computing and Content Centric Networking.

**Index Terms**—Security, Privacy, Sticky Policy.

## 1 Introduction

In an increasingly connected world, the security and privacy of the data transmitted across communication networks to enable value-added services represent a key concern for individuals, businesses and governments. While security and privacy have a long-standing tradition in terms of scientific discipline and technological domains, they have recently become a major topic in the public debate as well. Not a single day goes by without news about hacking of personal data, cyberattacks, scams and security breaches. As the awareness of the relevance of such issues raises, citizens and institutions become more and more concerned on how their data is communicated, transmitted and managed. Regulators are clearly entering the picture, through the establishment of legal frameworks for handling such aspects in a way able to protect the rights of data owners, ensuring sufficient guarantees for businesses to establish trust relationships and enable a true 'data economy'.

In this context, one hot topic concerns personal data, i.e., data about one identified individual or an individual who is reasonably identifiable by means of said data. An increasing amount of personal information is indeed continuously transmitted over the Internet. Domains and applications are extremely diverse, ranging from quantified self applications to purchases in retail chain stores, all the way to personal health records, sites visited while browsing the Web, information on current and historical locations etc. This does not apply only to consumers, but touches upon aspects at the very heart of enterprise life, such as customer relationship management data, financial projections and product life-cycle information. In all these cases, the communication of data is handled through technological infrastructures that need to provide appropriate control over the flow of the information themselves. And, of course, particular attention has to be paid towards the management of the security and privacy of said information [1]. Security and privacy management identifies the ways in which both organizations and individuals can control how (personal or sensitive) data are collected, accessed, transmitted, used and shared. A well-defined system for controlling the compliance of security and privacy policies and regulations would help organizations to improve the trust towards their customers. Such policies/regulations may involve the application of specific laws, national legislations, standards, specific processes or ad hoc restrictions.

In general, users have a rather poor understanding of how security and privacy regulations and technologies work (e.g., access authorization, sharing with third parties etc.). The main challenges include: (i) how to empower data producers so that they have more control over their information; (ii) how to gather and manage data producers' informed consent and, possibly, revocations; (iii) how to enforce technical measures for preserving data confidentiality and integrity, also when information is transmitted across multiple organizations/parties [2]. In fact, information often flows across different companies and, even, nations' boundaries. Further complexity arises because security and privacy laws may differ from one country to another one [3]. As a consequence, it becomes more difficult to enforce them or monitor compliance.

Current mechanisms and solutions for guaranteeing data protection and privacy preservation across organizational boundaries are built on frameworks based on contracts or service-level agreements, which can be further supported by means of enforcement and/or auditing mechanisms, in order to trace the actions that happen within the analysed system [4]. Advances in information and communication technology are leading to a huge growth in the amount of

- *D. Miorandi was with U-Hopper, via A. da Trento 38122 Trento, Italy.*
- *A. Rizzardi, S. Sicari and A. Coen-Porisini are with Università degli Studi dell'Insubria, via G. Mazzini 21100 Varese, Italy.*

*Manuscript received —— 00, 0000; revised —— 00, 0000.*

data flows, including personal and confidential information. The latter factor calls for the adoption of fine-grained security policies, down to the *per-user* and *per-data* level. This badly matches with the features of traditional approaches to security and privacy, whereby communication links (networks, servers) are secured by means of encryption and access control is managed by a centralised authority based on static policies.

Sticky policies [5], [6] were introduced some 15 years ago with the initial aim of providing a fine-grained user control on how their data gets disclosed and accessed. In the sticky policies approach, security policies are sent together with the data units, and follow them along their whole life-cycle (including when they cross administrative boundaries). Said security policies are tamper-proof and state the requirements/conditions to be applied to the data and ensure that appropriate constraints will be audited and put in action during each step of the information flow. A trust authority is required to check compliance of a given agent with the policy, and thereby deciding whether access should be granted. Sticky policies represent therefore an elegant, lightweight, and distributed solution to security and privacy issues. At the same time, they come with some inherent limitations (complexity, overhead, scalability) which have so far prevented their adoption at scale in real-world scenarios. Yet, with the arising of cloud computing and, more in general, of large-scale distributed ICT systems, researchers and practitioners have turned their attention again to sticky policies as a promising approach to solve issues emerging in such contexts. Some are even proposing that the ubiquitous adoption of sticky policies would solve many of the current security and privacy challenges.

In this paper, we aim to provide a complete survey on the state of the art of the current research and industrial efforts about the use of sticky policies for enforcing security and privacy in data disclosure and sharing. In this way, we aim at discussing the feasibility of proposed approaches, together with their robustness and efficiency, in order to reveal the open issues and draw some hints for future research directions and concrete applications. Our work complements relevant literature, and in particular [7], by providing a focussed discussion on the implications of the adoption of sticky policies in various contexts both from a research and an industrial point of view. Moreover, an accurate analysis of the critical aspects, as well as the advantages, towards the real deployment of sticky policy-based systems is carried out.

The remainder of the article is organized as follows. Section 2 describes the sticky policy paradigm and its application in the data security field, with particular attention to the open issues and to the impact of the current regulatory framework. Section 3 presents a survey of the technical solutions, based on sticky policies, available in the literature. Section 4 provides a discussion about the analysed approaches and points out further steps towards the development of new solutions in different application areas, while Section 5 concludes the paper by pointing out directions for future work.

## 2 Background and Motivation

In this section, we first point out the open issues in the current security and privacy management systems. Then, we introduce the sticky policy paradigm and explain their relevance with respect to the aforementioned open issues. In particular, we contextualize the background of the use of sticky policies and its implication for guaranteeing a high level of data confidentiality by enforcing proper rules.

### 2.1 Open issues

Guaranteeing data security and privacy represents a high-priority concern for many organizations. The importance of implementing proper security practices stays in the need for the companies to gain the trust of their market (e.g., e-commerce, social networks). Specific constraints are also imposed by regulations, which should be satisfied to guarantee certain levels of data protection, as presented in Section 2.3.

The implementation of robust and effective security and privacy mechanisms must face the following main issues [8]:

- Often business processes, in their early design phases, are conceived without considering security and privacy requirements; what happens is that lots of personally identifiable information are collected even if not strictly required to provide a given service;
- Many existing services adopt security and privacy systems that make use of more identifiable data than necessary, thus without avoiding the use of those which are not useful for complete the business processes;
- Companies are often unaware of the amount of sensitive data that they store;
- Companies are also often unaware about the security and privacy laws to be applied on stored and shared data and, at the same time, of the kind of consent the users gave to certain information at the time it was provided.

To cope with such issues, the following mechanisms for security and privacy management have been proposed in the literature [7]:

- Anonymization and encryption techniques;
- Access control systems;
- Policy management and enforcement.

Another crucial aspect is that of digital identities and profiles. Such kind of information may be used by organizations for various purposes: (i) improving and customizing the offered services; (ii) obtaining statistics for planning marketing strategies; (iii) selling them to third parties. In some cases, such a behaviour generates value for the end user, since it allows users to interact with and be aware of information of interest with other people, service providers, different companies and also government institutions. On the other side, privacy preservation for data owners is at serious risk, due to possible abuses and/or leakages of personal information.

Often, data owners have a very limited knowledge of the security and privacy policies applied to their personal or confidential information by different stakeholders, mainly in the case of information disclosure to third parties. In such situations, data owners have, in general, a very limited control over the future use of their sensitive and non-sensitive information, thus revealing an issue regarding the trust towards the involved parties [6].

Common examples are the cases in which users should read the so-called "terms and conditions" concerning the treatment of their information when disclosed to other companies/organizations (e.g., authorizations requested by web sites for performing various transactions). Such a task usually requires a lot of time, that users do not want to spend, and a level of comprehension of technical and legal aspects, which are often out of reach of lay users. However, when users give their consent, they enable the third parties to use the provided data in the way established by the aforementioned "terms and conditions".

To deal with such issues, new solutions for identity and privacy management should be put in place, aiming to simplify user experience and, at the same time, enforce the security policies applied to users' personal data, also by recognizing violations and malicious activities across multiple parties. To make it possible, users should be empowered with new mechanisms for handling a controlled disclosure of data and digital profiles not only just after the initial step, but also trusted third parties could be adopted and could act as intermediaries for the users in order to monitor if the information flow in compliance with the active policies.

At the moment, users and companies/organizations have little control over the enforcement of the policies to be applied on the data they provide. In particular, when a company/organization acquires information from their customers, it should guarantee that such data will be treated respecting the established regulations. This is made difficult also due to the complexity of the various systems taking part to the interactions and transactions among diverse platforms.

This is not only a matter of privacy for the users' data, but it also concerns how data are handled/shared, and which rules/policies are applied for the access to resource in situations where data exchange involves different application domains. The final aim is to ensure the compliance with the desired policies setting up a structure as distributed as possible in order to guarantee adequate performance in terms of computational load, processing, delays, response times, and storage. In this direction, sticky policies represent a feasible solution, due to their flexibility and capability of being used across different application contexts, as explained in the following. Moreover, they will be able to provide transparent policies for the users, thus overcoming the issues emerged above, mainly with regards to data confidentiality.

### 2.2 Sticky Policies

The sticky policy paradigm was proposed for the first time by Karjoth, Schunter, and Waidner in 2002 [8]. In sticky policies, conditions and constraints, that establish how data should be treated, are directly attached to the corresponding data. Sticky policies are able to regulate how data can be accessed and used throughout their entire life cycle, allowing access control decisions and policy enforcement to be carried out in a distributed fashion. In this way, a clever control over the protection of sensitive information, regarding both individuals and businesses, is reached. Such an approach has been mainly introduced for enforcing data confidentiality and further rules: when submitting information to a data consumer, a user/producer/owner consents to the applicable policies selecting the desired preferences.

Figures 1 and 2 sketch the difference between a "traditional" approach based on the transmission of data to a system regulated by access control policies and an approach based on sticky policies transmitted along with the associated data. In the former case, the owner of the data owns all the access permissions, the encryption keys and the credentials needed for identifying himself/herself and for ciphering the data to be sent to a particular company or data consumer; each data consumer that wants to decrypt such data has to own the related policies and credentials. In this case, the data owners have to trust the data consumer (or the data consumers) with which they share policies and credentials; they have also to trust that their data are collected and shared with other domains in compliance with the agreed rules. Note that such policies/credentials could be updated or revoked and, therefore, a system for their synchronization among all the involved data consumers must be put in place. Instead, in the latter case, data consumers do not own the policies/credentials: A trust authority is responsible for their management. The owner of the data sends them in an encrypted way along with the associated sticky policy; then the data consumers can contact the trust authority in order to obtain the access permissions on the received data. Therefore, in such a situation, data owners have to "trust" the trust authority itself, but are protected from illegitimate behaviours carried out by data consumers to fulfil their own purposes (as an example, marketing strategies, profit, and so on). Moreover, in this way, no synchronization is required among the involved data consumers.

The original approach of Karjoth, Schunter, and Waidner [8] presented some limitations, and requires significant improvements in terms of robustness and efficiency. In fact, it does not provide support for complex policy definition and the prevention or modification of the policies is not guaranteed.

Note that, in some scenarios, individuals or businesses confidential information may flow across different organizational boundaries. As an example, personal patients' profiles and data may be shared, through a service provider, by a healthcare system to pharmaceutical companies, or hospital specialists, or doctors belonging to other third parties. A similar example regards a travel agency, which may need to disclose data to car rental agencies, flight companies, or hotels for facilitating further reservations. Such scenarios are becoming more frequent with the arising of cloud computing, where users usually interact and disclose information with various service providers that often may further share these information with other service providers with the final aim of providing the required responses (e.g., in the form of a service). Another application context is that of the Internet of Things (IoT) [9], which concerns the interaction among heterogeneous entities and the provision of composite services; to this end, data from different sources

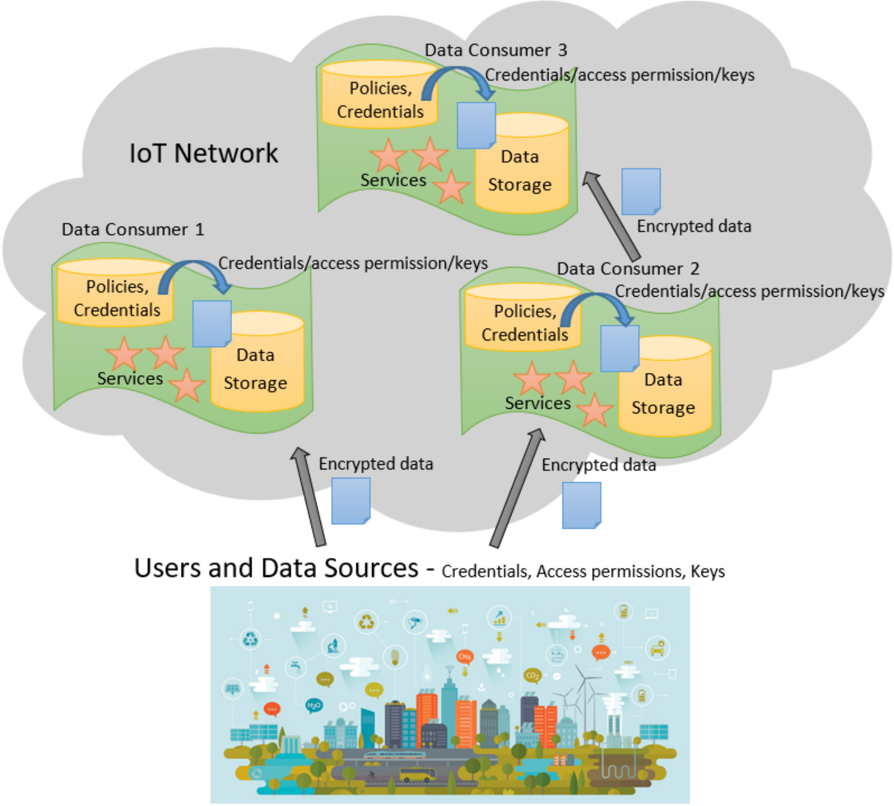

Fig. 1: Traditional approach

have to be merged, thus needing a share of information among multiple providers through a common platform [10]. Pearson and Mont [7] were the first to adopt sticky policies, within the EnCoRe project, for providing a mechanism for managing sensitive data across organizational boundaries in an accountable way. Section 2.4 will detail the application areas of interest for this new technology.

In all the presented cases, users or businesses must reveal personal/sensitive information to receive the desired services, but it is also important for them to control how such information are treated. Note that often services' vendors sell information collected from targeted users to statistical institutions or to companies interested in carrying on marketing campaigns. A popular example is represented by social networks, where users' personal data are sold to advertisers for making profit. In these situations, data owners should be able to directly control how their data are processed, managed, and disclosed by explicitly determining, in some ways, their preferences and policies, as just stated in Section 2.1. Such choices must be maintained along the entire data life-cycle, including the possibility to update or revoke them. Achieving this goal requires to propagate the stated policies to all the involved parties and deploying proper techniques to ensure that the policies are always observed [11]. Moreover, mechanisms able to trace the various interactions and transactions that happen within a system must be put in place, as well as policy fulfilment auditing. This would represent an added value, which would allow the users to have an in-depth control over the flow of their own information.

Specifically, sticky policies regulate the use of the associated data, and can define the following aspects:

- The owner of the data;
- The data content, possibly encrypted (see Section 3.1);
- The use to be made of the data (e.g., for marketing purposes, statistical analysis, transaction processing);
- Where and when data will be available (e.g., an expiration time-stamp, a certain area/company/set of companies);
- Specific obligations and restrictions for the parties involved.

An example of sticky policy, structured by means of XML tags, is presented in Listing 1.

```
<data>
<owner>the owner of the data</owner>
<encrypted data content>data encrypted with the
    adopted encryption mechanism</encrypted data
    content>
<policy>
  <use>allowed use for the data</use>
  <target>allowed use for the data</target>
  <validity>expiration timestamp</validity>
  <constraints>obligations and restrictions</
    constraints>
</policy>
</data>
```

Listing 1: XML sticky policy example

Furthermore, data anonymization techniques, notification of disclosure, or data elimination (e.g., after a certain time or after certain events) could be made available by

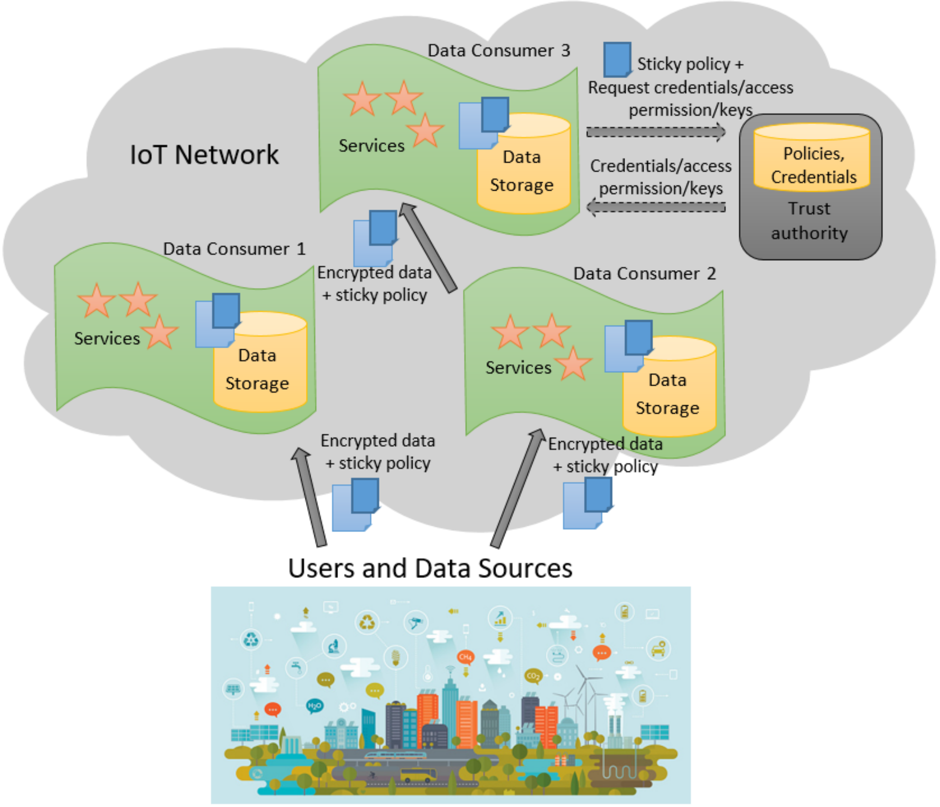
IoT Network
Data Consumer 3
Services
Data Storage
Sticky policy + Request credentials/access permission/keys
Credentials/access permission/keys
Policies, Credentials
Trust authority
Data Consumer 1
Services
Data Storage
Encrypted data + sticky policy
Data Consumer 2
Services
Data Storage
Encrypted data + sticky policy
Encrypted data + sticky policy
Users and Data Sources

Fig. 2: Sticky policy-based approach

the systems that manage the information or requested by the data owners themselves. Finally, the presence of a list of trusted authorities, able to ensure correct access to protected data, is also an essential requirement.

### 2.3 Regulations

The adoption and the definition of sticky policies must be compliant with the current legislation active in the various countries. Note that such regulations are usually derived from the privacy principles established by the *Organisation for Economic Cooperation and Development (OECD)* [12], which are the following:

1) *Collection Limitation Principle*: the acquisition and collection of personal/sensitive data should be limited as much as possible, and data owners should be aware about such data acquisitions as well as they should be able to give their consent or not
2) *Data Quality Principle*: the personal data should be acquired for specific purposes and their accuracy, completeness, timeliness and precision should be preserved
3) *Purpose Specification Principle and Use Limitation Principle*: at the time when data are collected, the purposes for which they are acquired should be immediately specified and maintained during time, especially in case of incompatibility with other purposes required by third parties which do not request the consent to the data owners
4) *Security Safeguards Principle*: certain levels of data protection should be guaranteed, in particular as regards confidentiality, integrity, unauthorized access
5) *Openness Principle*: proper systems for establishing, in a short time, the existence, the nature, and the purposes of collected personal/sensitive data should be made available, in order to be always aware of the developments, the practices and the policies applied to personal/sensitive data
6) *Individual Participation Principle*: data owners should be able, at any time, to be aware whether or not a certain system owns data relating to him/her; they also have the right to request that data related to him/her are deleted or modified
7) *Accountability Principle*: proper systems in charge of ensuring that the principles just presented are effectively put into effect should be provided.

Other directives regards the General Data Protection Regulation (GDPR) [13], approved by the EU Parliament in 2016. Its aim is to protect the EU citizens from privacy and data breaches in an world that is moving towards a totally data-centric approach in communications field. In particular, the GDPR states the data owner rights and legal requirements, as follows:

- Data owners must be notified of the breaches that occur towards their data;
- Data owners must be informed about whether or not personal data concerning them is processed, where and for what scope by the data consumers; moreover, data consumers are required to provide to the data

owners a copy of their personal data, free of charge, in an electronic format;

- Data that are no longer being relevant to original scopes for processing, or a data for which owners have withdrawn their consent must be erased (and so further dissemination must cease) by the data consumers and third parties; this aspect is known as "the right to be forgotten";
- The principle of "privacy by design" must be applied by the data consumers (e.g., companies, organizations) by implementing appropriate technical and organisational measures, in order to meet the requirements of GDPR and protect the rights of data owners.

Such directives will provide more transparency as regards the use of the data and also an empowerment of data owners. Moreover, these regulations will help organizations and companies to provide adequate security mechanisms for protecting their information. An ideal solution concerns the design and development of a system able to guarantee the correct application of the presented directives in an automatic manner; in this way, data producers' trust towards data consumers will be improved.

A first important step is the identification of the sensitive or personal data among those managed in a particular context. Then policies must be formalized using a proper language (e.g., a text version for the users/customers and a machine-readable version for the enforcement mechanism). Moreover, the data producer's consent has to be managed through an authorization system. As a further step, the policy must also be enforced in a way in which the access control system only grants actions authorized by the stated policy. Furthermore, the resulting obligations need to be enforced as well. Finally, an audit mechanism will enable to later control and verify if the information were properly handled and accessed. A summary schema is presented in Figure 3.

Note that the role of users is fundamental, since they are the first to react to privacy violations to their personal information [14]. As a consequence, companies/organizations are increasingly considering the adoption of new regulations and restrictions for the management of the data obtained by users, in order to contrast the lack of confidence that is spreading among the customers, for example in e-commerce. Essentially, companies/organizations have to demonstrate the reliability of the used privacy protection mechanisms by means of the establishment of well-defined policies, able to enforce users' preferences in terms of security and privacy, also in presence of multiple parties. This clarifies the importance of the analysis conducted hereby about the state of the art in the investigated field.

### 2.4 Application areas

There is also the need to reveal what are the areas of interest and the advantages for the application of sticky policy paradigm. Sticky policies could certainly ease the application of and compliance with security and privacy regulations (mainly with regards to information confidentiality) in many application scenarios and, in particular in the context of new and emerging technologies.

- *Cloud computing*. Cloud computing, which has been conceived as a way of sharing resources and data among devices on demand, could take advantage from the use of sticky policies for enabling users to directly control the access and the sharing rules regarding their information. Cloud computing and storage solutions are third party data centres that are located far from the users' action range. As a consequence, users has to accept the treatment conditions when they use cloud services/applications or provide personal data. Naturally, cloud based systems offer proper levels of security and privacy [15], as well as policies, that regulate the disclosure of the managed information. However, users have to trust the service providers [16], which, in turn, can access the data that is in the cloud at any time, delete or modify them (also accidentally), and share information with third parties, even without the agreement of the users. To cope with such issues and prevent unauthorized access, users can encrypt the data before providing them to the cloud system, but often this is not possible or sufficient. This is due to the fact that the adopted encryption schema (also those based on homomorphic encryption) or the chosen encryption key may be weak and, thus, they may be easily compromised. Moreover, keys have to be kept on a separate storage block with respect to the encrypted data, as well as, a system for the backup of the keys is needed, in case such a storage block is attacked. Also the keys should automatically expire after a period of time and a refresh schedule should be handled. Obviously, these aspects require, from the cloud management perspective, an further effort. Last but not least, information taken from a cloud can be also sent to users' mobile devices, which, in turn, has to own the decryption keys, if the data are encrypted, in order to access them. This may be not desirable since mobile devices are overly vulnerable in terms of security. In this scenario, sticky policies could help users to pursue their rights and actively manage the rules to be applied to their own information. In fact, if cloud based systems would allow the users to provide not only the data but also the associated policies, then the system and the third parties should only be equipped with the necessary software for respecting the defined policies. A sort of a wrapper or parser layer should be adopted by the involved platforms in order to be compliant with the forced regulation. This represents a fundamental step for improving the trustworthiness of the cloud customers with respect to the actual service providers and for avoiding improper resource disclosure.
- *Internet of Things*. Similar problems arise in the Internet of Things context. In this case, the policy management is further complicated by the fact that heterogeneous devices are involved. Therefore, the difficulty emerges in the sense that a solution has to be designed to accommodate the different technologies and standards involved. Another important issue is related to the treatment of information within the IoT global network. In fact, the huge amount

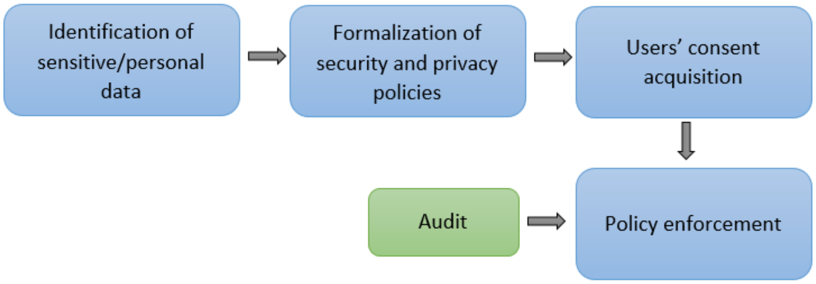
Identification of sensitive/personal data
Formalization of security and privacy policies
Users' consent acquisition
Policy enforcement
Audit

Fig. 3: Schema for the definition of a security mechanism

of data producers and data consumers puts in light that proper regulations must be adopted in order to manage the privacy of information from the time when they are acquired to their transmission among different endpoints [17]. The actual systems do not clarify in a transparent way how and with whom data are shared, thus leading to have no confidence/trust (from the data producers) in providing sensitive information to the IoT network (i.e., to the data consumers) [18]. As an example, sensitive information and users' habits can be derived from the analysis of the data obtained by fitness trackers, health devices, and so on. Such kinds of information may be used by business companies to make targeted offers or by hackers for performing some malicious activities. Nevertheless, sticky policies would allow to exercise a clever control over the access authorization/permission on IoT resources, for example by the data sources. Note that an environment, like that of IoT [19], is generally characterized by the presence of multiple applications domains and transfers of data from one realm to another. A conventional approach would not ensure that a data which passes from a domain application to another (or among different endpoints) is treated with the same rules, but this goal can be achieved by means of a mechanism based on sticky policies.

- *Context Centric Networks*. Another example is that of Context Centric Networks (CCN) [20]. In CCN, at a low network level, information are routed on the basis of their contents instead of their IP addresses. Data are securely stored on various hosts (i.e., multiple copies of the same data can be cached in different places). Moreover, data are identified by three properties: validity, provenance and relevance; such properties have the final aim of providing to the users a sort of reliability level as regards the received contents. In fact, a content-based security approach is adopted in CCN paradigm; in this way, protection and trust are associated to the content itself during data flow, and not to the connections over which it is transmitted during the time. All contents are authenticated by means of digital signatures, and confidential ones are encrypted. Instead, traditional IP networks based the trust of the contents on where (i.e., from what host) and on how (i.e., over what kind of connection) they have been gathered; therefore, data consumers, in order to trust the content, must directly retrieve it from the original source; while CCN provides an end-to-end security mechanism between the owner of the content and the consumer, who must evaluate by himself/herself if the received content is trustworthy or not. Embodying security in content, as CCN does, and not in the hosts, reduces the trust to be placed in the various network intermediaries. A further step in this direction would be the integration with sticky policy, in order to cope with several open issues [21], such as the redefinition of a proper privacy model and how to manage inter-domain policies.

Note that the contexts just presented are not considered trusted, since such kinds of platform are designed to be shared among multiple users and among different application domains, in which multiple stakeholders can act. In few words, they are not under full control of the users who provide their own data, but they are managed by third parties, and users have to trust them, as specified in Section 2.2. Such a main feature magnifies the need of designing and developing new mechanisms to handle and apply correct and transparent policies to the information that reside far from the user devices (e.g., cloud systems, IoT platforms, hosts of the CCN), in order to improve the trust towards the platforms by the data consumers' perspective.

## 2.5 Sticky policies in the real world

All major IT companies and industries are increasingly investing into the cloud and into the adoption of IoT technologies. A relevant feature in this field is represented by the acquisition and sharing of information in real-time, which would allow interested users/companies promptly take strategic decisions. Furthermore, real-time analysis provides valuable advantages in terms of quickly response to environmental changes. As an example, recent investments have been made to monitor the resources consumed by industrial plants in order to regulate production's flow and reduce costs (e.g., electricity, heating, water). To execute such control activities, various sensors can be disposed in the monitored area, but an ad-hoc infrastructure (e.g., an IoT network) is required for acquiring, processing, and storing (e.g., by means of a cloud) such data and actuating proper actions in response to the analysis' results. Note that information may be collected within the private network of the industry, but, in some situations, they may transit across the public network (Internet, cloud) for further purposes.

Hence, malicious entities may violate data confidentiality and integrity, or information may be accessed by unauthorized users. To prevent such threats, encryption mechanisms and policies must be directly applied to data. Security and privacy countermeasures must also preserve the system's efficiency, therefore lightweight and distributed solutions must be conceived. Note that services/applications that use insecure or incorrect data do not provide any benefits, since decisions taken on the basis of errors might negatively impact on finance or productivity. Therefore, as just said before, the market and technological expansions are leading to new issues to be addressed for guaranteeing an efficient and secure data management, not only to gain the trust of the consumers, but also to better manage the IT company/organization itself and reduce the activities' costs.

As a proof that the most important IT companies are more and more involved in cybersecurity issues, we can cite some relevant products designed and developed by IBM[1] and Cisco[2]. Their target is mainly cloud and mobile applications, but they also generally refer to application and network security, database protection, identity and access management. Instead, the industry's interest in new security management systems through sticky policy is witnessed by the recent diffusion of some pilot projects and prototypes. Such mechanisms will perfectly fit the principles of the emerging industry 4.0.

For example, the HP Laboratories worked on the development of a full working prototype for improving the accountability of systems such as e-commerce and B2B [6]. It is based on the following technologies: (i) a trusted platform, defined by the industry alliance *Trusted Computing Platform Alliance (TCPA)* and available on the market supplied by IBM, in charge of manage the network resources and transactions; (ii) an optimised version of the IBE encryption technique, which presents performance comparable to RSA; (iii) sticky policies for performing access control enforcement over users' data and guaranteeing privacy.

A project conducted within the *EPSRC (Engineering and Physical Sciences Research Council)* led to the definition of APIs and protocols for the cloud, able to provide fine-grained access control towards the cloud resources, as well as the enforcement of user-defined sticky policies independently from the resource's location [22]. The project had Cisco as a partner and such APIs and protocols were expected to be standardized by OASIS. Moreover, pilot implementations were planned for being distributed as part of the *OpenStack* suite of software, that involves more than 135 organizations, such as HP, Cisco, and Intel. The most interesting feature of this work is that it considers the case of data aggregation and split. More in detail, if data are merged with others or, vice versa, filtered or reduced, the policies associated to them must change accordingly to fit the requirements of the new data. How to derive the new sticky policy from the original ones still remains an open issue, but the project in [22] tried to develop a new algebra and algorithms to obtain the new sticky policy, by means of an ontology and previously associated meta-data.

Furthermore, some of the projects financed by the *EU 7th Framework Programme* adhered to the use of sticky policies, mainly for guaranteeing privacy over users' personal data. As an example, the *PrimeLife* project aimed to protect privacy in emerging Internet applications, such as collaborative scenarios and virtual communities, maintaining, at the same time, a life-long control of the privacy on data [23]. Enforcement and accountability systems have been based on the use of sticky policies, for which a human readable translation has been provided. Such a feature allowed users to verify and check the status of their personal identifiable information. The translation is performed in a standard way according to a defined ontology. The definition of an ontology, as we will discuss in the following, represents another of the main challenge about the sticky policy adoption.

Also the *EnCore* project [24] used sticky policies to represent and enforce the consent and revocation preferences of end-users. Note that sticky policies are not applied within the single organization, but they are generated and propagated only if information are disclosed to third parties. Then, such an approach is applied recursively across chains of organizations, and also updates/revocations are allowed and managed.

More recently, attention towards sticky policy paradigm has been paid within the *H2020* programme, by the *SPECIAL* project [25]. Its aim is to realize a technical solution which allows users/organizations to share big data, guaranteeing, at the same time, data protection compliance in a privacy-aware manner. Note that *SPECIAL* is built on top of the results obtained by the [23] project and will provide experiments on real case studies. Hence, it could represent a promising step ahead in the real application use of sticky policies.

The authors developed a prototype of an IoT middleware, named *NOS* (NetwOrked Smart object), which has been integrated with an enforcement framework based on sticky policies. We refer to [26] for more details about the architecture and implementation. Another middleware, named *ShareIff*, for Android has been proposed in [27]. It provides an API for the end-to-end secure sharing and display of self-destructing messages. More in detail, *ShareIff* makes available to the apps the possibility to encrypt a message on the sender's endpoint/device and send it to the recipient; the recipient, in turn, will be able to decrypt the message, which will be securely displayed only on the recipient's device, only for the amount of time specified by the sender. This is obtained by means of a sticky policy architecture. In such a case, the sticky policy contains the visualization time, which cannot be altered by external programs.

The work in [28] makes use of sticky policies to protect data access accountability and non-repudiation in the sharing of Office Open XML (OOXML) packages. Identity Based Encryption (IBE) primitive is integrated to securely bind the sticky policy and the data together.

## 3 Sticky Policy Solutions

In the following sections we survey and analyse the solutions available in the literature and present the state of the research in the field. The analysed works are clustered

[1] $https : //www.ibm.com/security/products$

[2] $https : //www.cisco.com/c/en_uk/products/security/index.html$

according to their target usage: policy enforcement in enterprises, policy languages, access control, and cloud systems. Before that, a section is devoted to discussing the encryption mechanisms proposed for sticky policies.

### 3.1 Encryption mechanisms applied to sticky policies

Depending on the level of robustness required by a specific application domain, data as well as sticky policies might be encrypted, and the access to the content must be allowed only according to the encryption strategy used. The most popular techniques proposed for sticky policies are: Public-Key Encryption (PKE), Identity-Based Encryption (IBE), Attribute-Based Encryption (ABE), and Proxy Re-Encryption (PRE). [29] provides an overview of the enforcement of sticky policies using such encryption mechanisms. They can be compared by considering different features, such as: (i) the degree of depth of security and privacy policies; (ii) the degree of complexity of updating/revoking them; (iii) the complexity of encryption keys' generation and distribution; (iv) the requirements of the trusted third party and of the policy enforcement system.

With regards to policy enforcement, we have that, using PKE, the data owner encrypts the content and the sticky policies with the public key of the receiver; in this way, it is the policy enforcement framework that enforces other constraints (e.g., access control preferences, data filtering). Instead, by means of IBE, the data owner encrypts the content and the sticky policies with the IBE identity of the receiver; such an identity acts as a public key and may be represented by any string. This is an innovative aspect of the emerging IBE cryptographic schema. For example, it may be a username, as in the case of PKE, to be concatenated with other constraints (e.g., a privacy preferences, roles) or keywords. In this case the policy enforcement framework acts as before; while the trusted third party is in charge of managing the release of the receiver's private key. Note that, the computation of the IBE decryption keys is usually performed by a trusted third party on the fly or can be postponed.

PKE and other systems based on public/private keys, such as RSA, make available the same functionalities provided by IBE, in terms of personal data obfuscation. As an example, a symmetric key may be generated by a user and further encrypted with a X.509 identity certificate, provided by a trust authority; hence, users' confidential information may be encrypted with such a key coupled with a hash value, obtained by the associated sticky policies. The use of the hash, in fact, represents a sort of digital signature, which provides a further security level. However, the schema proposed by IBE has two advantages: (i) it is more simple; (ii) it presents better scalability in presence of multiple trusted third parties with respect to traditional public/private keys approaches.

After this necessary clarification, we now introduce the ABE and PRE approaches. By means of ABE, the data owner encrypts the content and the sticky policies on the basis of an access structure, defined using attributes established by the specific system/domain; instead the policy enforcement framework is in charge of enforcing other constraints (depending on the specific application domain). PRE provides a general framework able to incorporate other encryption techniques (e.g., PKE, IBE, and so on). More in detail, the data owner (acting with the role of delegator) encrypts his/her contents using both his/her public key and another encryption key, assigned to the sticky policy; whereas, the policy enforcement system (acting with the role of delegatee) enforces other constraints (e.g., privacy preferences, access control policies). Different encryption schemes can be adopted by delegators and delegatees.

What emerges is that IBE guarantees a deeper level of flexibility and expressiveness with respect to the adoption of PKE within an enforcement framework, which, on the other side, provides a less complex keys' management. [29] pointed out that the enforcement systems that use PRE allow to adopt multiple encryption techniques at the same time, thus improving the usability of sticky policy approach.

### 3.2 Sticky policy enforcement in enterprise applications

Enforcing users' privacy by giving more control to data owners and, at the same time, increasing the trust in companies/organizations, is the goal of the approach proposed in [6] and [30]. Such a solution is presented referring to an e-commerce application. In this scenario, profile information and identity are transmitted by a user to a certain e-commerce site in order to obtain a requested service/product. As a consequence, an on-line profile of the user is created and the user can login, after giving the consent to the security and privacy policies to be activated on the provided data. However, the e-commerce site may require to contact other web sites or organisations for obtaining other useful information, thus disclosing user personal data to such third parties. In such a situation, the trust model put in action by the e-commerce site and the user does not include other external entities, highlighting the problem of data sharing and of the users' privacy. Users should be able to maintain the control over their personal data in a simple and intuitive way, also across different domains. To this end, the authors propose a theoretical approach (a running prototype is still under development) based on sticky policies, able to trace the service provision chain by means of two enabling technologies: IBE and TCPA (Trusted Computing Platform Alliance). TCPA is responsible for checking the integrity of computer platforms and their installed software; while sticky policies and data are encrypted using IBE schema. Hence, if the data content or the IBE identity is altered or tampered, the trust authority will not be able to generate the correct decryption key, thus the data would not be accessible. The presence of the trust authority avoids the need to exchange secrets among users and web sites/organizations when personal data are transmitted; while the use and the format of sticky policies allows to select which data should be disclosed depending on the user identity associated to the IBE encryption keys. To this end, an XML-based representation of policies is adopted [6], enabling the expression of multiple types of constraints, permissions, or obligations. Summarizing, the contribution of [6] relies in the mechanisms provided to couple tamper-resistant disclosure polices (i.e., IBE identity along with sticky policies) to personal/sensitive data; in this context, web sites/organizations are forced to respect the established policies on the basis of users' attributes and, moreover, the are also traced and checked by means of a proper system (i.e., TCPA).

As an extension of [6], [7] details a project, named EnCoRe, aimed at enabling users to establish their consent to security and privacy policies and to update/revoke them at any time, by means of a proper Web-based GUI. In particular, EnCoRe adopts sticky policies to propagate and enforce the preferences of the data owners by means of consent and revocation. A flexible toolbox solution was developed, which can be deployed and customized by the interested organizations/service providers. The sticky policies sent by the EnCoRe system to the companies/organizations, involved in a transaction specify not only the purposes of data processing, but also the related permissions or other tasks to be completed (e.g., deletion after the expiration of a certain time-stamp, revocation, and so on), previously stated by the data owners. The trust authority is conceived as a shared entity because the spreading of information, associated with the sticky policies, among multiple parties, is monitored and controlled by an external component of the EnCoRe system. The main task to be performed by the trust authority is to translate the high-level requirements expressed in the sticky policies into fine-grained access policies/permissions. Note that such an approach is similar to that of [6]. Both an evaluation of the purposes for which personal/sensitive data are required and a risk assessment are carried out in order to establish if release the data or not. An important aspect is that the work proposed in [7] provides a mechanism for the automatic propagation of users' preferences throughout the system and across the companies/organizations involved. PKI is used for encrypting the transmitted information. Along with the proposed solution, the authors also reveal many open issues in the analysed research area, such as: the integration with a solid and distributed enforcement framework, an effective risk assessment functionality, and auditing mechanisms. The final aim is to guarantee that organizations can ensure the correct application of policies independently from where the data are processed.

The EC TAS3 project [31] has developed a policy enforcement infrastructure able to support the transfer among different sites of personal/sensitive information together with their sticky policies. The system is able of handling multiple policy languages, and has a master policy decision point (PDP) that will resolve any policy decision conflicts. Obligations are used to guarantee that the sites that owns the data will attach the related sticky policies before sending them to other sites, that will also store the data along with the sticky policies. Information are safely put in the policy store, which is in charge of managing the mapping among data and the associated sticky policies. Such a mapping is many-to-many since a policy may be applied to more than one data and, viceversa, a data may have many policies associated to it. To this end, a global identifier is assigned to each sticky policy, in order to be used across different realms. If the policy store is trusted, then the policies could be placed there without further security mechanisms. Instead, if the store is not trusted, such policies must be protected against tampering by means of encryption mechanisms or digital signatures.

A novel approach, named Type-based PRE (TPRE), which extends the original PRE, presented in Section 3.1, is adopted in a general framework with the aim of enhancing sticky policy enforcement [29]. Such a solution starts from the assumption that data owners may be required to select, among different Policy Enforcement Points (PEPs), the one to use for enforcing his/her data policies, on the basis of certain security and privacy requirements. Such requirements may depend on the degree of sensitiveness of the treated information and, thus, a data owner will decide to pay more or less money according to the trustworthiness of the chosen PEP. Therefore, the proposed enforcement mechanism provides a good level of flexibility to the data owners empowering them with the capability of independently establishing at which level to enforce the policies associated to their data. With respect to the original PRE approach, this one maintains all the previous advantages and, in addition, it requires only one pair of public/private key, instead of multiple ones. Moreover, if a PEP is compromised, it can be replaced by other PEPs, without disrupting the entire system. Note that, in such a scenario, all PEPs can access the managed data, because they are all in charge of enforcing the associated policies. However, data and policies need to be encrypted in order to mitigate the vulnerability of the system to external attacks. The comparison results, as presented in Section 3.1, indicated that PRE provides better performance in terms of policies and keys update/revocation with respect to the other presented encryption mechanisms. The authors of [29] extended it with TPRE with the final goal of improving its performance in terms of key management efficiency (i.e., by reducing the use of multiple keys for the different sensitive levels associated to the data), as just said above.

[14] and [32] also describe an approach towards the enforcement of security and privacy guarantees within organizations/companies. Indeed, the proposed framework aims at allowing organizations/companies to publish transparent security and privacy policies to the interested users, thus enabling them to give their consent and also specify some preferences before their data are collected and processed. Sticky policies are adopted because they perfectly fit the customer centric model considered by the authors. Nevertheless, the policy enforcement is carried out only within the organization/company and this represents the main drawback of such an approach. The innovation put in place in [14] and [32] is the concept of "version" of the security and privacy policies. In fact, if a policy previously established by the organization/company is later updated, the users' data collected before must be managed according to the policy active at the consent time, and not to the new one, without the consent of the users themselves. Only if the user gives his/her consent to the application of the new policy, then his/her data can be processed according to the new rules.

IBM proposed to its customers an Enterprise Privacy Architecture (EPA), which represents a methodology for companies/organizations to provide transparent and well-defined levels of security and privacy [8] and, at the same time, to fully exploit the personal information collected for business purposes. The main feature of such a solution is that security and privacy policies can be customized depending on the needs of the specific company/organization. EPA is composed of four main blocks, which are: (i) the analyser of security and privacy regulation, which is responsible for identifying the current applicable regulations, also across different realms; (ii) the management reference model, which allows a company/organization to establish and enforce a specific security and privacy strategy put in act within the company/organization itself; (iii) the security and privacy agreement framework, which consists in a methodology for finding the existing links connecting business processes to users or departments of the company/organization or also to third parties; in this way, it is possible to trace and monitor the activities taking place within the company/organization; (iv) the technical reference architecture, which defines the technology needed for implementing and deploying the defined policies.

The Platform for Enterprise Privacy Practices (E-P3P) is an improvement of the EPA's technical reference architecture [5]. E-P3P separates the deployment of policies within a specific company/organization from the mechanisms that control the entire life cycle of the collected data. Keeping these two aspects decoupled fosters the respect of the established regulations, which are further extended by associating a consent policy to each collected information. In this way, a finer-grained access control is achieved, even down to the per-person level, and able to support versioning of policies (i.e., to account for policies updated over time), the definition of roles for the users, and, even, to handle different legislation (e.g., Europe and US), thus among diverse companies/organizations.

Some questions emerge from this first group of solutions tailored to enterprise-level applications. More in detail:

- How to couple the sticky policies management with an efficient key management system?
- How to efficiently define a hierarchy of the sticky policies adopted in an enterprise's application?
- How to support sticky policy synchronization flow and update among different enterprises with low overhead and delay? Is it also possible to support multiple versions of the same policies?

### 3.3 Sticky policy languages

Along with EPA, which is, as discussed above, an enterprise architecture, IBM also proposed a policy language, named Enterprise Privacy Authorization Language (EPAL). It is based on the concept of policy refinement, since it allows to check if a company's/organization's regulation fulfils the preferences set by the customers/users or adheres to the standards established by laws. In fact, the security and privacy policies adopted within a company/organization should comply with both internal regulations and legal constraints. The sticky policy paradigm could be realized hereby and would address the secure exchange of data from one realm to others in a secure and privacy-preserving manner and, also, to deal with the problem of checking whether a policy refines others. [33] presents an effective algorithm able to deal with such an issue by means of the IBM EPAL. More in detail, a policy defined using EPAL is composed by: (i) a vocabulary, which determines the usable conditions, purposes, users' hierarchies, and obligations; (ii) a list of authorization rules; (iii) a global condition; (iv) a default ruling. The algorithm presented in [33] checks the policy refinement, concerning the current security and privacy regulations, analysing the active conditional rules and potential inheritances among permissions/obligations.

Even with regards to EPAL, the authors of [34] state that it does not fully cope with situations where information along with security and privacy policies flow across different applications. The sticky policy approach can help in this direction, forcing to apply the correct policies also when governed by other realms, but they are not currently supported by many existing applications' interfaces. Therefore, [34] introduces a framework able to support and monitor the entire security and privacy policy life cycle, identifying the main technological aspects as well as non-technical components, and highlighting the relationships among them. Substantially, the work in [34] points out some useful hints for future research towards the deployment of robust solutions in policy management field.

The work in [35] distinguishes between access control, that states the conditions to be fulfilled before a data is disclosed to a particular requestor, and usage control, that states how data has to be treated and processed once they are disclosed. With regards to usage control for privacy, it is relevant in downstream usage, where proper restrictions and regulations should be put in place on the data to be shared. To this end, a two-sided XML-based policy language is presented in [35]. It allows data owners and data consumers to express their preferences: the former in terms of allowed paths to permit the flow of information and the related usage restrictions; the latter in terms data treatment policies. Note that downstream usage paths can be specified at any desired depth. Moreover, a matching algorithm is described, by which users can promptly verify whether all hops in a proposed policy path match their own preferences, allowing them to decide autonomously and in an automated way whether it is safe to disclose the requested personal/sensitive information. When a match occurs, a sticky policy is generated describing the correct rights and obligations that the data consumers have to adhere to. In the authors' opinion, the adoption of EPAL and P3P policy languages represents a promising solution. Unfortunately, EPAL and P3P (i.e., the language used for the E-P3P architecture, presented in Section 3.2) are not very expressive in the case of personal information sharing with third parties or in case of downstream usage. In particular, EPAL, as just said above, founds the definition of policies on vocabularies established within a specific company/organization, thus not considering downstream usage. Whereas, P3P makes available a limited number of classes of third-parties to which information can be shared, leaving to the server the responsibility of managing and classifying other third-parties in one or more of these classes.

Another solution may make use of different languages, as the work described in [36] suggests. Such a work is the result of a multidisciplinary project which involved both lawyers and computer scientists with two main goals: (i) defining proper motivations behind the need of security and privacy data protection; (ii) analysing and designing proper protection mechanisms combining both legal and technical aspects. In this direction, the authors of [36] give importance to provide appropriate tools for managing users' consent before personal data disclosure (in this case, by means of a friendly interface). A framework is proposed, which delivers the consent given on data through software agents. Different languages, such as SIMPL (which is used for simplify policy definition by users), have been proposed, because, in the authors' opinion, they are needed for representing the variety of entities that acts in the analysed scenario (i.e., users, agents). The choice should take into account the following principles: (i) the most appropriate language should be adopted for each purpose; (ii) minimal sets of terms and conditions should be used; (iii) a mapping between different languages should be accurately stated. The compliance with such principles would ensure that policies will be consistent in the presence of multiple stakeholders. Furthermore, the framework is also responsible for resolving possible policy conflicts regarding the treatment of personal/sensitive data performed by agents.

The fact that the use of sticky policies is just proposed in standard languages, such as the IBM EPAL, highlights their growing importance. However, their usage in general-purpose authorization scenarios is still limited, apart from the framework, named PeerAccess, proposed in [37]. PeerAccess represents a distributed system, able to assess access authorizations. All the involved parties can sign the data contents before sending them to others. The owner of the data dictates the criteria about its disclosure to other peers. Such criteria will be valid also in the case of data passing to third parties. However, the problem of what type of information could be derived from the original contents is also to be tackled, which current sticky policies approaches do not overcome. Hence, PeerAccess is extended in order to monitor and control the disclosure of derived information. In some cases, some non-identifiable information can be also released without the consent of the data owner. What emerges is the need of a logic that is able to classify the various kinds of information and, then, make an evaluation about their disclosure, on the basis of the associated sticky policies. As an example, the data related to the number of occurrences of a disease in a specific county may be released with the consent of the patients; from such information, the number of occurrences of that disease in the whole state may be derived and disclosed, also without the patients' consent. Summarizing, the scope of the presented approach is to enable its application to multiple policy languages, as a sort of wrapper.

Finally, what lacks in the approaches presented above is that they perform an a posteriori verification over the correct application of the policies; while mechanisms for acquiring logs would be useful for assessing the accountability of the parties involved in the various transactions. The authors of [38] discuss about the information to be included into logs, by means of a well-defined language, named the PrimeLife Privacy Policy Language (PPL), to support a posteriori analysis. PPL is derived from the well-known XACML representation language. Users and data consumers will express in a symmetric way the required access and usage control rules, thus simplifying the policy management. Until now, a log compliance analyser has been designed and implemented, along with an abstract version of a subset of PPL. They have been demonstrated the importance of the log design for accountability exploiting real use cases.

Summarizing, what emerges from such an analysis is that no common agreement exists on:

- Which language better expresses the desired security and privacy policies?
- Which level of depth the policies have to able to represent?
- How to derive the resulting sticky policy from the original ones in case data are aggregated or splitted?

### 3.4 Sticky policy for access control

Recently, several researches suggested that, for efficiently enforcing security and privacy policies on data managed within companies/organizations, such policies must be integrated into novel access control systems. For such a reason, the authors of [39] extended the well-known Role Based Access Control (RBAC) model with the new Purpose-aware RBAC (PuRBAC) model. It associates fine-grained access control permissions not only to the roles of the users, but also considering their purpose for accessing data and, possibly, further constrains (e.g., time validity, conditions). Hence, several purposes can be assigned to different roles, and authorization will be determined on the basis of these two parameters. In this way, policy administration is simplified by the fact that purposes are conceived as separate entities and, thus, the generation of complex policies is avoided. As a final achievement, the enabling for an automatic reasoning about the actual purposes with respect to users' performed tasks would be a valuable extension for guaranteeing better reliability to the system. In the authors' opinion, a very flexible approach to deploy such security and privacy policies is represented by the sticky policy paradigm, because they are suitable to the policies' structure just defined in the sense that different rules can be attached to different instances of the same data type. However, such an approach seems to be complex for being adopted by companies/organizations. In fact, the main drawback, mentioned in [39], is that policies are just stuck to data, so that companies/organizations will lose their centralized control over access control procedures. Furthermore, authors believe that the use of sticky policies will require a huge amount of storage and computational resources with respect to a traditional access control approach.

In the field of ubiquitous computing, [40] focuses its analysis on the emerging access control methodologies aimed at guaranteeing more robustness for security and privacy. Related technologies are surveyed and further classified by the authors in three categories, corresponding to the three access control phases: (i) prevention; (ii) avoidance; (iii) detection. The authors also distinguish the life-cycle of personal/sensitive data into other three phases, which are: (i) collection; (ii) access; (iii) secondary use. Moreover, the authors point out that traditional access control models, such as RBAC, DAC (Discretionary Access Control) and MAC (Mandatory Access Control), are not adequate in the field of ubiquitous computing, because they are conceived as centralized approaches. Ubiquitous computing requires indeed a distributed solution, able to handle the dynamic relationships among users and data. In fact, ubiquitous computing involves a huge amount of computers' interactions and, as a consequence, services and resources should be accessed in a distributed and efficient manner. In such a scenario, security and privacy represent two key pillars due to the amount of personal/sensitive data, such as activities performed by the users, users' location, and so on, transmitted by sensors or other devices, in order to provide the requested services to the interested users. Conventional authorization systems are not secure enough to support such a heterogeneity. As a solution, sticky policy paradigm could be adopted and coupled, as the authors suggest, to the IBE encryption schema. In fact, by means of sticky policies, access regulations travel along with the corresponding data in a distributed system and, at the same time, audit mechanisms could be deployed in order to reveal potential attacks and trace the system's behaviour against both authorized access and violations.

Also theoretical models have been proposed. For example, in [41], the authors describe a theoretical model for security and privacy assessment/control in context-aware systems. The security and privacy model is constructed by means of information related to surrounding spaces. Such data can be used to realize a security and privacy control based on relevant contextual factors, available resources, offered services, and other similar ones. Information types are grouped into privacy tags and assigned to sets of owners in charge of establishing the associated access permissions. Note that a privacy tag includes the following parts: (i) a space's handle, which determines, for each object belonging to the space, its type; (ii) a policy, which specifies the active permissions related to the operations allowed for that type of information; it consists of a set of security and privacy controls, aimed to express the data owners' rights and the enabled data readers; (iii) a privacy property list, which describes some relevant features of the information, such as accuracy, lifetime, level of confidence, and so on. The authors also discuss how decentralization can be introduced, which is a valuable property mainly in pervasive computing systems, using unified privacy tagging. Note that the authors of [41] state that sticky policy approach, as that used in IBM EPA, often limits the computations that can be performed on metadata regarding read and write operations; for such a reason, they decided to introduce their own tags.

Therefore, in the works related to access control, sticky policies are understood as not sufficiently flexible and poorly performing. Traditional access control models are also criticized, thus leading to the introduction of novel solutions, based on new access control models or data tagging. Therefore, the questions which emerge are:

- New solutions for the enforcement of security and privacy policies should be based on traditional access control models or new ones should be pointed out?
- Are sticky policies able to cope with the requirements of robustness and efficiency required by the actual access control systems?

### 3.5 Sticky policy in the cloud

Nowadays, cloud computing and virtualization are becoming the de facto standard choice for companies/organizations or private individuals for storing and managing data. What actually lacks is the trust in such a kind of infrastructures, as pointed out in Section 2.4. The main reason lies in the lack of a transparent declaration of terms and conditions applied to the storage and processing of the data. In fact, cloud customers do not fully control their data and do not have clear technical guarantees on such aspects. To cope with such an issue, a security service for the cloud, named SPACE, is introduced in [42]. It enables data owners to directly specify their security and privacy preference by means of proper tools. What is interesting is that SPACE exploits sticky policies to offer access and usage control functionalities in a distributed manner over the cloud. In this way, users are aware of how their personal/sensitive information are stored and treated at any time.

Instead, the work in [43] points out that three approaches may be adopted for applying sticky policies in the cloud: (i) the PKI-based approach; (ii) the IBE techniques; (iii) the secret sharing techniques. The choice will depend on the context and on the trust model involved, but the authors claim that the one based on the secret shares would provide fine-grained control and better performance in terms of computation, storage, and transmission bandwidth with respect to the other solutions. In fact, using secret sharing instead of a public key infrastructure would simplify the key management when multiple parties are involved.

Personal health records (PHR) contain a significant amount of sensitive information. Online personal health records services enable individuals to create, store, manage, and share his/her personal health data in a centralized way. In this manner, the PHR owners are able to access third-party health care services and monitor their health status. The usage of cloud computing services for storing and processing PHRs raise a number of rather obvious security and privacy concerns. A commonly adopted solution to protect information is encryption. Basically, the PHR owner encrypts her PHR data by herself before uploading them to the cloud, where it also stores the cyphertexts. Only the parties trusted by the PHR owner can be authorized to acquire the decryption keys and then decrypt the cyphertexts. Contrarily, for the unauthorized parties, who do not have the corresponding decryption keys, the PHR data remains confidential. However, encryption alone is not sufficient for guaranteeing a fully secure system: it is also required to enforce fine-grained access control on the managed information, according to the agreed policies, which should be specified by the PHR owner. Such policies are based on: the kind of the PHR data, the role a user plays, the privileges to read or write, the purpose of using the data, the time and location conditions, obligations/restrictions, and so on. To make sure that the security and privacy-aware access control policies he/she specified are enforced, the PHR owner can stick these policies to his/her PHR data and then upload the data associated with the sticky policies to the cloud, using PRE encryption schema. Unfortunately, the cloud server is semi-trusted by the PHR owner, therefore it cannot directly enforce the policies. Therefore, the trusted third parties should be selected by the PHR owner in order to be compliant with the satisfaction of users' requirements specified in the sticky policies. In [44], the authors sketch a scheme to enable the protection of the PHR data hosted in the cloud. It not only supports fine-grained data access, but also an effective encryption mechanism and a flexible key management approach to enforce the policies sticky to the PHR data. Hence, the PHR owner can, at her own will, select new encryption keys for her PHR data (thus changing their access control policy) or delete her PHR data when required. Multiple trust authorities are responsible for verifying the compliance/enforcement of the sticky policies associated to the PHR data and, consequently authorize or not the requesting user to acquire the decryption key for accessing the data itself. An innovative aspect is that trust authorities are not allowed to know the decryption key of the PHR data (as happens in [6], described above). Through implementation and simulation, [44] shows that the scheme is both efficient and scalable. Note that there is no need for a central policy repository because the policies are always stuck to the data. The policy may be represented in any format, such as XML or JSON.

Still in healthcare context, blockchain [45] and sticky policy might be married to securely manage healthcare data, as pointed out in [46]. In such a work, an architecture based on blockchain is coupled with a purpose-centric access model; an untrusted third-party is in charge of processing patients' data computation without violating privacy, following the MPC (Secure Multi-Party Computing) principle. As stated by the authors, the sharing of healthcare data is important for providing smarter and high-quality healthcare services. In this context, patient could retain control on their (sensitive) healthcare data, and thus preserve their privacy, even if information is scattered in different healthcare systems (e.g., hospitals, pharmacies, and so on). It is worth to note that users have not to trust any third-party, since the blockchain systems is fully responsible for guaranteeing privacy.

Another field of interest for the application of sticky policies is that of big data. A framework for guaranteeing security in big data applications has been proposed in [47], adopting the sticky policy paradigm. In this solution, data chunks and related sticky policies are separately stored in order to improve the reliability and the robustness of the whole system. The proposed architecture is divided into two logical components: (i) the trust authority domain, which consists of further two elements: an identity/key management engine, which manages the users, the keys, the authorizations and the associated privileges, and a policy engine, which is the core of the trusted authority domain because it controls all the accesses to data, also tracing the involved parties; (ii) the data centre domain, responsible for storing encrypted data. The data centre domain may be trusted if the underlying cloud is private; whereas, it may be untrusted if the underlying cloud is public or hybrid. In the former case, the trusted authority domain and data centre domain can be logically deployed together without the need of data encryption; while in the latter case, data must be encrypted for avoiding to expose the system to serious attacks. The IBE encryption schema is recommended by the authors. Note that, if data are encrypted, then they can only be released when specific conditions, determined by the policies are satisfied. Trust authority is in charge of checking the users' credentials before disclosing the data decryption key. At the same time, is has to audits and logs the operations of access authorization and restrictions executed within the system.

Within the cloud, the work in [48] focuses on dealing with jurisdictional privacy and security risks by proposing a technical solution able to demonstrate the reliability of the mechanisms of data protection. Hence, they introduce the figure of a "corporate entity", which is responsible for evaluating and proposing contracts to be associated to the services offered by the cloud, both for a private use and to be offered to customers. In this scenario, sticky policies are only used for controlling the release of data to third parties and notifications to data owners.

We can conclude that cloud-based systems present many open issues in terms of how to provide user control on the usage of their sensitive data. More in detail, some questions arise about the encryption mechanism to be used, as well as the adoption of a proper key management scheme. The existing solutions try to address such problems making available robust encryption mechanisms associated to the sticky policies, along with the capability of update and revoke them at any time. But:

- How to customize the desired behaviour depending on the specific application domain?
- Besides logging the access control and policy disclosure operations, how to react to violation attempts?
- Could blockchain, coupled with sticky policies, be a feasible mechanism for ensuring trust along the whole data lifecycle (i.e., during the whole information flow)? Is this technology mature enough in terms of security to be adopted in IoT, cloud, CCN, and other emerging Internet-based systems?

### 3.6 Other applications

In the previous sections, solutions based on sticky policies have been analysed in different application domains. In this section, other scenarios are presented, related to mobile systems, parallel jobs, and digital ecosystems.

Mobile devices manage every day, in different locations and situations, a huge amount of data, which are often sensitive ones and, thus, need to be kept confidential. A prototype, named ProtectMe, based on sticky policies, has been proposed in [49] in order to attach proper usage conditions to the information sent in mobile contexts. Constraints are derived from users' preferences and from contextual information, directly acquired by mobile devices. In this manner, users are assisted in the access control policies' definition. Therefore, ProtectMe provides a user-friendly support to data consumers with the final aim of simplifying the management of access control operations and, at the same time, of proving the feasibility of the use of sticky policies in mobile applications.

The work in [50] aims at coordinating, in an efficient manner, the execution of multiple parallel jobs in a memory cache. In such a context, sticky policies are adopted for avoiding incomplete files to be stored in a file system. If an incomplete file is recognized in cache, it is stuck (by means of a proper policy) to be blocked until it is totally removed from cache, thus preventing its further use. As a consequence, corrupted or incomplete files, which are not useful for jobs yet, are prohibited and do not allow complete files to be properly used.

A new emerging and innovative paradigm in the field of dynamic business integration is that of digital ecosystems, where federations of companies, organizations or institutions can be created with the aim of collaborating or compete regarding specific topics. In this distributed context, the main challenge is the management of identities. Several solutions have already been proposed in the literature, but they have proved not to be sufficiently flexible for such a dynamic environment. More in detail, a comprehensive solution should consider four important aspects: (i) companies/organizations/institutions adopt different kinds of non-compatible certificates and identity mechanisms (e.g., Kerberos, X.509, SPKI), thus a standards should be defined; (ii) different administrative domains, located in diverse places, may be involved and, as a consequence, users need to access/interact to/with one or more of their offered services/applications; (iii) how to share identities within the various federations, taking into account that a company/organization/institution may join or leave a federation at any time; (iv) how to define a proper trust model. To deal with such issues, [51] proposes an identity management model able to process, in an automated manner, the identities among the components of the digital ecosystem. This model is based on the OASIS Security Assertion Markup Language (SAML v2.0) standard, which provides support for guaranteeing the interoperability among the different existing technologies. Users profiles are defined instead of integrating a more complex distributed identity storage. Then, user profiles are peer-to-peer replicated on trusted nodes, thus decentralizing their management. No hierarchy of users is considered in the presented solution. Access to the users' profiles is regulated by means of sticky policies, exploiting the mechanisms of Access Control Lists (ACL). More in detail, when a user enters in a new session, his/her profile is downloaded on a secure memory (that depends on the used device) and then decrypted to be processed on the basis of the scope of the application. Once the session ends, the user's profile is encrypted and updated on the current trusted peer, to be further replicated to other ones.

This section presents interesting applications of the sticky policy paradigm in emerging scenarios. But:

- Is it possible to expand the use of sticky policies in mobile applications in order to address the actual security issues in short-range information sharing?
- The idea of identity profiles provided for the digital ecosystems could be adapted for other scenarios (e.g., mobile devices, social networks)?

# 4 Discussion

A comparison of the approaches presented in Section 3 is proposed hereby, in order to highlight what are the most common techniques adopted in terms of encryption, languages and architectural styles. Then, pros and cons of the sticky policy paradigm versus a "traditional" approach are analysed. Finally, relevant topics for future research directions are discussed.

## 4.1 Comparison of the existing solutions

Many open issues emerged from the analysis carried out in Section 3. First of all, as far as the encryption mechanisms (detailed in Section 3.1) are concerned, we point out that many solutions do not specify any technique to be used for ciphering the data associated to the sticky policies. However, this is a crucial requirement for protecting not only the data but also the policies themselves from being tampered or violated by malicious entities. Table 1 collects all the works presented in Section 3, which are classified on the basis of the used encryption mechanism. As just said, the most part of the considered solutions makes no use of an encryption technique or, at least, it is not explicitly specified.

TABLE 1: Encryption mechanisms

| Encryption technique | Work |
|---|---|
| *PKE* | [7], [51], [43] |
| *IBE* | [6], [30], [40], [47], [43] |
| *ABE* | – |
| *PRE* | [29], [44] |
| *secret sharing* | [43] |
| *not specified* | [31], [14], [32], [33], [34], [8], [5], [35], [36], [37] [52], [38], [39], [41], [42], [48], [49], [50], [46] |

Another important remark must be done about the languages adopted for the specification of sticky policies, since its expressiveness influences the level of depth that can be associated to the policies themselves. A widely-accepted language still does not exist, but two proposals seem to emerge: XML and EPAL. Note that the works which adopt XML criticize EPAL and vice versa, as it happens in [35] and [41]. Table 2 highlights the languages used by the solutions of Section 3, distinguishing in works which adopt: (i) XML formalism; (ii) EPAL language; (iii) multiple languages, as specified in Section 3; Table 2 also represents the works which do not specify any language for policies' definition. The existence and adoption of a commonly-accepted policy specification language is another fundamental requirement for guaranteeing interoperability and the correct enforcement of policies among different systems/organizations.

TABLE 2: Policy languages

| Language | Work |
|---|---|
| *XML* | [6], [30], [35], [44] |
| *EPAL* | [33], [34], [8], [5], [38] |
| *multiple languages* | [31], [36], [37], [52], [51] |
| *not specified* | [7], [29], [14], [32], [39], [40], [41], [42], [47], [48], [49], [50], [43], [46] |

Moreover, as regards the presented approaches, it is important to distinguish between distributed and centralised solutions. The former ones present advantages for the application in heterogeneous and large-scale contexts (e.g., ubiquitous computing, digital ecosystems, big data), where many entities need to interact in real time and, therefore, need quick response about service requests and access permissions. In this direction there is already a discrete number of solutions, as presented in Table 3. Other ones, instead, are still based on a centralized architecture, often making use of a central trust authority or on a centralized enforcement framework.

TABLE 3: Architectural approaches

| Approach | Work |
|---|---|
| *distributed* | [7], [29], [33], [34], [37] [52], [40], [41], [42], [44], [43], [46] [47], [48], [51] |
| *not distributed* | [31], [35], [36], [39], [49], [50] |
| *not clear* | [6], [30], [14], [32], [8], [5], [38] |

### 4.2 Pros and cons of sticky policies

On the basis of what emerged in this discussion, the advantages of adopting a system based on sticky policies, with respect to a "traditional" approach, can be summarized as follows:

- The owners of the data have more control over them and, in particular, they can establish specific rules about who can access the data and which authorizations have to be allowed, also when information flow across different realms;
- The flow of the data among different application domains or businesses can be controlled in a more effective way; in other words, the whole life-cycle of the use of a data can be monitored from its generation to its transmissions and disclosure;
- A possible reduction of the policies' management costs for the organizations, companies or businesses, since third parties will be in charge of setting up the policy enforcement system. In fact, as depicted in Figure 2, one or more trust authorities could have the responsibility of supervising all access requests and permissions;
- The capability, for data owners, of performing an offline management of the policies because they are attached to the data/resources, thus they must not be necessarily accessed in an online mode;
- An overall improvement of the reliability of the system in terms of confidentiality, preventing unauthorized data disclosure to not trusty third parties.

Besides these pros, there are various shortcomings, which can be detailed as follows:

- It is difficult to establish an adequate set of policies; in this direction, it would be useful to define an ontology or a taxonomy for the policies in a way that the involved parties may agree on a set of standard ones. Note that the definition of a taxonomy or an ontology includes two aspects: (i) the classification of the policies; (ii) their semantics. Both such features must be accomplished to obtain a structured and well-defined policy vocabulary.
- As a consequence, a difficult in finding a standard arises, thus it also emerges the need of defining interpreters for the policies' dictionary established at the previous point;
- Due to the fact that the policies are no longer stored into local storage units managed by the platforms/-companies/organizations, but policies are transmitted into the network along with the data, this naturally causes an increase in the amount of information transmitted. Such an increase in the use of the bandwidth could compromise the network performance (e.g., in terms of delay), mainly in environments characterized by the transmission of huge amounts of data in real time, such as IoT. Such an issue must be taken into account in the definition of a new system based on sticky policies. Note that this also implies an increased computation complexity in the processing of the data chunks at each transmission step.
- Since the owners of the data have also to manage the sticky policies to be associated to the data, then the devices they use for interacting with the network may be affected by a higher computational load. Such an issue becomes more relevant in environments where constrained devices could act, such as IoT.
- Mainly in cloud based systems and CCN, resources are redundant; this means that many copies of the same information may be stored/cached in different hosts belonging to the system. This naturally points out the problem of policy propagation in the different parts of the system itself.

Such revealed shortcomings may have caused the slow diffusion of the use of sticky policies to guarantee security and privacy in real systems. In fact, putting in act an enforcement framework based on sticky policies would require the presence of very qualified professional figures in the field of security, able to set up the network architecture in a proper way, as well as to define the access control rules in compliance with the needs of the interested organizations. Issues which can emerge may be related to possible increments in bandwidth requirements, or other possible overheads in the information transmission. Moreover, the data will no longer be internally managed by the organizations themselves, but a third party will be in charge of performing data management. In that sense, the role of the third party becomes fundamental and must be accepted by the involved organizations. Another important issue is how to integrate such a new approach in existing access control systems; in fact, organizations may disagree on completely reorganizing their whole network architecture and already working security mechanisms. Moreover, from a commercial viewpoint, the main issue is that service/infrastructure vendors/providers may not push for the development of systems implementing sticky policies, given the lack of a well-established market for such solutions. The adoption is also hampered by the lack of a set of well-defined standards for the representation and adoption of sticky policies [53], a pre-condition for interoperability and for big players to invest in the area.

On the other side, researchers are very interested in sticky policies due to their strong fallout the policy level, as they would allow to federate and/or integrate/orchestrate third-party services in a privacy-preserving end-to-end manner. It is also worth remarking that, in the presence of systems under the full control of a single administrative entities, traditional approaches (such as RBAC/ABAC) are sufficient, and sticky policies may not be required; where sticky policies become valuables are situations in which a plurality of entities need to exchange data in a controlled way. Examples of such situations are business ecosystems, in which companies linked by business contracts need to exchange data relative to their interactions (think, e.g., of a supply chain and its actors) or personal data platforms, where individuals may be willing to share personal data in a controlled way to digital platform operators. In order to cope with such shortcomings, a further effort by researchers and industries in developing working prototypes and applications of sticky policies in real case studies, as presented in Section 2.5, would help other organizations in increase their confidence in the effectiveness of this kind of solution for improving security and privacy management.

### 4.3 Research directions

Based on the elements identified in the previous sections, we hereby provide a list of what we believe are the main challenges to be tackled for sticky policy to be deployed at large. They cover a wide range of aspects, from the more scientific ones all the way to practices and standardization. We do hope that the list can be of interest for various audiences, including most notably PhD candidates, research consortia and IT industry.

- **Algebra for sticky policies.** A common theme in the IoT world is how to handle privacy and access control when different data streams get aggregated or processed (e.g., downsampled or obfuscated). This would call for the definition of an algebra for handling in an automated and formally controllable way such operations. The sticky policies should therefore support a set of operators, which would represent the transformation the data goes through.
- **Anonymization.** To provide anonymization techniques is another fundamental aspect for guaranteeing data confidentiality. To this end, the adoption of a suitable encryption scheme would allow to efficiently transmit, in a ciphered way, both the data and the associated sticky policy. Some promising approaches have been surveyed in Section 3.1 and could represent relevant starting points for further investigations in the area. Besides the encryption technique (which should take into account the possibility of dealing with power-constrained devices), also an efficient key management system should be devised. With respect to key management, several aspects should be defined, ranging from key distribution across different application realms all the way to key update and revocation.
- **Incremental deployment and integration with legacy access control.** In many cases sticky policies will be introduced in systems which already employ some form of access control. How to handle transitions? How to incrementally deploy sticky policies while the ’old’ (traditional) access control system is still in place? This requires some non-trivial engineering steps for ensuring a peaceful coexistence of both systems during the transition phase.
- **Porting access control schemes to sticky policies.** In many environments access control schemes are already defined, based on roles (e.g. RBAC) or, more general, attributes (ABAC). An interesting aspect (somehow related to the previous challenge) is to understand how to port such schemes to sticky policies. Take the ABAC case; traditionally ABAC is based on associating resources with a set of metadata, and access permissions are based on such metadata. In the sticky policy case there is no need to define metadata, as the policy itself is used to define access rules, which then can be enforced based on attributes of the user/service who wants to access them. Of course there are plenty of details to be introduced, but having some methods for the automated porting of existing access control schemes is something that can have a major impact on the adoption of sticky policies.
- **Policies Propagation and Synchronization.** In a real environment, policies may change over time. This can reflect, e.g., a change in the policy specification language or a change mandated by the actual data controller. In a distributed, large-scale settings, propagating such changes and ensuring system-level coherence may be hard to achieve. This calls for the definition of management plane mechanisms able to distribute and synchronize policies across the whole system in a controllable way. To cope with such issues, some frameworks provide mechanisms for enabling the use of a versioning scheme for policies (e.g., [14] and [32]), or for guaranteeing the possibly of revoking (e.g., [7]) or updating (e.g., [47], [51]) the policies after a certain interval time, but a perfectly fitting solution has yet to be determined.
- **Partially trusted systems and blockchain.** Sticky policy-based systems typically require the presence of a trusted party in order to manage access permissions to resources. Can we envision a system in which parties are only partially trusted (e.g., honest-but-curious)? Can we envision decentralized solutions which work even in the presence of malicious parties? Can blockchains or similar distributed ledger technologies be applicable to such scenario? Such questions represent open research challenges which, while probably having low impact in the short term, could significantly boost the chances of sticky policies being adopted at scale in the long term
- **Monitoring and Attack Detection.** Monitoring represents a crucial requirement for any security system, in order to detect attacks and check the ability of the system to behave as expected. In the case of sticky policy, such monitoring should extend however to the complete data life-cycle. And as a data item may cross multiple administrative domains throughout its lifetime, the monitoring functionality should be

able to follow the whole data usage trail. This represents a distinctive feature with respect to traditional monitoring and detection solutions. In this respect recent approaches in provenance [54] could represent an interesting starting point.

- **Performance and Low-Overhead Solutions.** One of the drawbacks of sticky policies (in particular in the IoT field) relates to the overhead they naturally create, which increases the bandwidth requirements. It would be desirable to envisage schemes for low-overhead sticky policies, which would lower the entry barrier for the adoption of such schemes in resource-constrained environments.
- **Standardization.** A standardization process for sticky policie is definitely needed. This should include: (i) the establishment of an ontology or a taxonomy able to include all the terms, the expressions and the proper syntax for the definition of sticky policies; (ii) the definition and specification of a proper representation language for access permissions within sticky policies, following the defined ontology/taxonomy; (iii) the definition of a format for the sharing of sticky policies. In particular, the ontology/taxonomy definition should take into account and satisfy the global regulations relating to data protection and should also be able to work across multiple domains.

All such aspects are waiting to be analysed by the research community, in order to design and develop new solutions for enabling the use of sticky policies to improve the resilience and the security of the actual systems and the data consumers' trustworthiness.

## 5 Conclusion

The paper has presented the sticky policy paradigm as one of the emerging technique for improving security and privacy aspects in Internet-based systems. A survey of the state of the art about the actual use of sticky policies for improving the confidentiality in case of data sharing among different data consumers has been analyzed. Many open challenges have been pointed out, thus shedding some light on research and industrial directions in this field. Addressing such relevant problems would allow sticky policy to evolve and guarantee high levels of security and privacy in various application domains and in IT industries.

## References

[1] C. Bettini and D. Riboni, "Privacy protection in pervasive systems: State of the art and technical challenges," *Pervasive and Mobile Computing*, vol. 17, Part B, pp. 159 – 174, 2015.

[2] B.-J. Koops, J.-H. Hoepman, and R. Leenes, "Open-source intelligence and privacy by design," *Computer Law & Security Review*, vol. 29, no. 6, pp. 676 – 688, 2013.

[3] S. Corbett, "The retention of personal information online: A call for international regulation of privacy law," *Computer Law & Security Review*, vol. 29, no. 3, pp. 246 – 254, 2013.

[4] G. Gurkaynak, I. Yilmaz, and N. P. Taskiran, "Protecting the communication: Data protection and security measures under telecommunications regulations in the digital age," *Computer Law & Security Review*, vol. 30, no. 2, pp. 179 – 189, 2014.

[5] G. Karjoth, M. Schunter, and M. Waidner, "Platform for enterprise privacy practices: Privacy-enabled management of customer data," in *International Workshop on Privacy Enhancing Technologies*. Springer, 2002, pp. 69–84.

[6] M. C. Mont, S. Pearson, and P. Bramhall, "Towards accountable management of identity and privacy: Sticky policies and enforceable tracing services," in *Database and Expert Systems Applications, 2003. Proceedings. 14th International Workshop on*. IEEE, 2003, pp. 377–382.

[7] S. Pearson and M. C. Mont, "Sticky policies: An approach for managing privacy across multiple parties," *Computer*, vol. 44, no. 9, pp. 60–68, 2011.

[8] G. Karjoth, M. Schunter, and M. Waidner, "Privacy-enabled services for enterprises," in *Database and Expert Systems Applications, 2002. Proceedings. 13th International Workshop on*. IEEE, 2002, pp. 483–487.

[9] S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini, "Security, privacy and trust in internet of things: The road ahead," *Computer Networks*, vol. 76, pp. 146–164, 2015.

[10] S. Sicari, A. Rizzardi, D. Miorandi, C. Cappiello, and A. Coen-Porisini, "A secure and quality-aware prototypical architecture for the internet of things," *Information Systems*, vol. 58, pp. 43–55, 2016.

[11] S. Sicari, A. Rizzardi, D. Miorandi, C. Cappiello, and A. Coen-Porisini, "Security policy enforcement for networked smart objects," *Computer Networks*, vol. 108, pp. 133 – 147, 2016.

[12] "http://www.oecd.org/sti/ieconomy/oecdguidelinesontheprotectionofprivacyandtransborderflowsofpersonaldata.htm," *OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data*, 2016.

[13] "http://www.eugdpr.org/," *European GDPR*, 2017.

[14] P. Ashley, C. Powers, and M. Schunter, "From privacy promises to privacy management: a new approach for enforcing privacy throughout an enterprise," in *Proceedings of the 2002 workshop on New security paradigms*. ACM, 2002, pp. 43–50.

[15] S. Singh, Y.-S. Jeong, and J. H. Park, "A survey on cloud computing security: Issues, threats, and solutions," *Journal of Network and Computer Applications*, vol. 75, pp. 200–222, 2016.

[16] I. M. Abbadi and A. Martin, "Trust in the cloud," *Information Security Technical Report*, vol. 16, no. 3-4, pp. 108–114, 2011.

[17] R. H. Weber, "Internet of things: Privacy issues revisited," *Computer Law & Security Review*, vol. 31, no. 5, pp. 618–627, 2015.

[18] Z. Yan, P. Zhang, and A. V. Vasilakos, "A survey on trust management for internet of things," *Journal of Network and Computer Applications*, vol. 42, pp. 120–134, 2014.

[19] A. Ouaddah, H. Mousannif, A. A. Elkalam, and A. A. Ouahman, "Access control in the internet of things: Big challenges and new opportunities," *Computer Networks*, vol. 112, pp. 237–262, 2017.

[20] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*. New York, NY, USA: ACM, 2009, pp. 1–12.

[21] A. Ghodsi, S. Shenker, T. Koponen, A. Singla, B. Raghavan, and J. Wilcox, "Information-centric networking: Seeing the forest for the trees," in *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*. New York, NY, USA: ACM, 2011, pp. 1:1–1:6.

[22] "http://gow.epsrc.ac.uk/ngboviewgrant.aspx?grantref=ep/j020354/1," *Sticky Policy Based Open Source Security APIs for the Cloud*, 2012.

[23] "http://www.primelife.eu/," *PrimeLife*, 2011.

[24] "www.encore-project.info," *EnCore project*, 2012.

[25] "https://www.specialprivacy.eu," *SPECIAL*, 2017.

[26] S. Sicari, A. Rizzardi, D. Miorandi, and A. Coen-Porisini, "Security towards the edge: Sticky policy enforcement for networked smart objects," *Information Systems*, vol. 71, pp. 78–89, 2017.

[27] A. Goulão, N. O. Duarte, and N. Santos, "Shareiff: A sticky policy middleware for self-destructing messages in android applications," in *Reliable Distributed Systems (SRDS), 2016 IEEE 35th Symposium on*, 2016, pp. 11–20.

[28] G. Spyra, W. J. Buchanan, and E. Ekonomou, "Blockchain and git repositories for sticky policies protected ooxml." Vancouver, Canada, 2017.

[29] Q. Tang, "On using encryption techniques to enhance sticky policies enforcement," 2008.

[30] M. C. Mont, S. Pearson, and P. Bramhall, *Towards Accountable Management of Privacy and Identity Information*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 146–161.

[31] D. Chadwick and K. Fatema, "Distributed privacy policy enforcement by using sticky policies," *W3C Workshop on Privacy and data usage control*, vol. 13, no. 13, p. 14, 2010.

[32] C. S. Powers, P. Ashley, and M. Schunter, "Privacy promises, access control, and privacy management. enforcing privacy throughout an enterprise by extending access control," in *Electronic Commerce, 2002. Proceedings. Third International Symposium on*, 2002, pp. 13–21.

[33] M. Backes, G. Karjoth, W. Bagga, and M. Schunter, "Efficient comparison of enterprise privacy policies," in *Proceedings of the 2004 ACM symposium on Applied computing*. ACM, 2004, pp. 375–382.

[34] A. I. Antón, E. Bertino, N. Li, and T. Yu, "A roadmap for comprehensive online privacy policy management," *Communications of the ACM*, vol. 50, no. 7, pp. 109–116, 2007.

[35] L. Bussard, G. Neven, and F.-S. Preiss, "Downstream usage control," in *Policies for Distributed Systems and Networks (POLICY), 2010 IEEE International Symposium on*. IEEE, 2010, pp. 22–29.

[36] D. Le Métayer, "A formal privacy management framework," in *International Workshop on Formal Aspects in Security and Trust*. Springer, 2008, pp. 162–176.

[37] M. Winslett, C. C. Zhang, and P. A. Bonatti, "Peeraccess: A logic for distributed authorization," in *Proceedings of the 12th ACM Conference on Computer and Communications Security*, ser. CCS '05. New York, NY, USA: ACM, 2005, pp. 168–179.

[38] D. Butin, M. Chicote, and D. Le Métayer, "Log design for accountability," in *Security and Privacy Workshops (SPW), 2013 IEEE*. IEEE, 2013, pp. 1–7.

[39] A. Masoumzadeh and J. B. Joshi, "Purbac: Purpose-aware role-based access control," in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*. Springer, 2008, pp. 1104–1121.

[40] S. Yamada and E. Kamioka, "Access control for security and privacy in ubiquitous computing environments," *IEICE transactions on communications*, vol. 88, no. 3, pp. 846–856, 2005.

[41] X. Jiang and J. A. Landay, "Modeling privacy control in context-aware systems," *IEEE Pervasive computing*, vol. 1, no. 3, pp. 59–63, 2002.

[42] S. Trabelsi and J. Sendor, "Sticky policies for data control in the cloud," in *Privacy, Security and Trust (PST), 2012 Tenth Annual International Conference on*, July 2012, pp. 75–80.

[43] M. Beiter, M. C. Mont, L. Chen, and S. Pearson, "End-to-end policy based encryption techniques for multi-party data management," *Computer Standards & Interfaces*, vol. 36, no. 4, pp. 689–703, 2014.

[44] C. Leng, H. Yu, J. Wang, and J. Huang, "Securing personal health records in the cloud by enforcing sticky policies," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 11, no. 4, pp. 2200–2208, 2013.

[45] I.-C. Lin and T.-C. Liao, "A survey of blockchain security issues and challenges," *IJ Network Security*, vol. 19, no. 5, pp. 653–659, 2017.

[46] X. Yue, H. Wang, D. Jin, M. Li, and W. Jiang, "Healthcare data gateways: Found healthcare intelligence on blockchain with novel privacy risk control," *Journal of Medical Systems*, vol. 40, no. 10, p. 218, Aug 2016.

[47] S. Li, T. Zhang, J. Gao, and Y. Park, "A sticky policy framework for big data security," in *Big Data Computing Service and Applications (BigDataService), 2015 IEEE First International Conference on*. IEEE, 2015, pp. 130–137.

[48] S. Pearson and A. Charlesworth, "Accountability as a way forward for privacy protection in the cloud," in *IEEE International Conference on Cloud Computing*. Springer, 2009, pp. 131–144.

[49] F. Di Cerbo, S. Trabelsi, T. Steingruber, G. Dodero, and M. Bezzi, "Sticky policies for mobile devices," in *Proceedings of the 18th ACM Symposium on Access Control Models and Technologies*, ser. SACMAT '13. New York, NY, USA: ACM, 2013, pp. 257–260.

[50] G. Ananthanarayanan, A. Ghodsi, A. Wang, D. Borthakur, S. Kandula, S. Shenker, and I. Stoica, "Pacman: coordinated memory caching for parallel jobs," in *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*. USENIX Association, 2012, pp. 20–20.

[51] H. Koshutanski, M. Ion, and L. Telesca, "Distributed identity management model for digital ecosystems," in *The International Conference on Emerging Security Information, Systems, and Technologies (SECUREWARE 2007)*. IEEE, 2007, pp. 132–138.

[52] S. Bandhakavi, C. C. Zhang, and M. Winslett, "Super-sticky and declassifiable release policies for flexible information dissemination control," in *Proceedings of the 5th ACM workshop on Privacy in electronic society*. ACM, 2006, pp. 51–58.

[53] "www.etsi.org/deliver/etsi_gs/ins/001_099/005/01.01.01_60/gs_ins005v010101p.pdf," *Identity and access management for Networks and Services; Requirements of an Enforcement Framework in a Distributed Environment*, vol. ETSI, 2011.

[54] L. Moreau, P. Groth, J. Cheney, T. Lebo, and S. Miles, "The rationale of prov," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 35, pp. 235–257, 2015.

**Daniele Miorandi** Daniele Miorandi is Executive VP R&D at U-Hopper. He received a PhD in Communications Engineering from Univ. of Padova, Italy, in 2005. His current research interests include modelling and performance analysis of large-scale networked systems, ICT platforms for socio-technical systems and distributed optimisation for smart grids. Dr. Miorandi has co-authored more than 120 papers in internationally refereed journals and conferences. He serves on the Steering Committee of various international events, for some of which he was a co-founder (Autonomics and ValueTools). He also serves on the TPC of leading conferences in the networking and computing fields. He is a member of ACM, ISOC and ICST.

**Alessandra Rizzardi** Dr. Alessandra Rizzardi received her BS and MS degree in Computer Science 110/110 cum laude at the University of Insubria in Varese (Italy), in October 2011 and July 2013, respectively. In December 2016 she got her Ph.D. in Computer Science and Computational Mathematics at DiSTA (Department of Theoretical and Applied Science) in Varese (Italy), under the guidance of Prof. Sabrina Sicari. Her research activity, conducted in the group of Prof. Alberto Coen Porisini and Prof. Sabrina Sicari, is focused on wireless sensor networks, wireless multimedia sensor networks and IoT (Internet of Things) security and privacy issues. Now she is a postdoc researcher in Software Engineer at the University of Insubria, in the same research group. She has been author of more than 15 scientific papers which have been published in international journals and conference proceedings.

**Sabrina Sicari** Sabrina Sicari is Associate Professor. She received her laurea degree in Electronical Engineering,110/110 cum laude, from University of Catania, Catania, Italy, in 2002. In March 2006 she got her Ph.D. in Computer and Telecommunications Engineering at the same university, under the guidance of Prof. Aurelio La Corte. From September 2004 to March 2006 she has been a Visiting Scholar at Dipartimento di Elettronica e Informatica, Politecnico di Milano, Italy under the guidance of Prof. Carlo Ghezzi. Since May 2006 she works at Dipartimento di Informatica e Comunicazione, UniversitÃ  degli Studi dell'Insubria in software engineering group (head Prof. Alberto Coen-Porisini). Since 2008 she is a member of Computer Networks (Elsevier) editorial board. Since 2016 she is also a member of IEEE IoT editorial board and since 2017 she is a member of the editorial board of both ETT and ITL. In 2012 she has been a guest editor of Ad Hoc (Elsevier) special issue "Security, privacy and trust in Internet of Thigs era" (SePrit).

**Alberto Coen-Porisini** Alberto Coen Porisini received his Dr. Eng. degree and Ph.D in Computer Engineering from Politecnico di Milano (Italy) in 1987 and 1992, respectively. He is Professor of Software Engineering at Universita' degli Studi dell'Insubria (Italy) since 2001, Dean of the the School of Science from 2006 and Dean of the Universita' degli Studi dell'Insubria since 2012. Prior to that he was Associated Professor at Universita' degli Studi di Lecce (1998-2001), Assistant Professor at Politecnico di Milano (1993-2001) and Visiting Researcher with the Computer Security Group at University of California, Santa Barbara (1992-1993). His main research interests are in the field of specification and design of real-time systems, privacy models and wireless sensor networks.