Preserving Privacy in a P2P Social Network

Monica Mordonini, Agostino Poggi, and Michele Tomaiuolo(✉)

Department of Information Engineering, University of Parma, Parma, Italy
michele.tomaiuolo@unipr.it

Abstract. Building centralized social networking systems has many drawbacks, e.g., lack of privacy, lack of anonymity, risks of censorship, and operating costs. As it is discussed in this article, an alternative approach is possible. A prototype system, named Blogracy, has been realized as a micro-blogging social networking system, based on well-known P2P technologies, such as DHTs and BitTorrent. In particular, this article presents the security architecture of the system, which relies on a key-based identity system and a scheme for attribute-based content encryption, with multiple authorities. Moreover, some empirical results obtained in test operations over PlanetLab are presented, comparing plain and I2P anonymized communications.

Keywords: Social network · Peer-to-Peer · File sharing · Distributed hash table · Anonymity · Confidentiality · Key-based identity · Attribute-Based Encryption

1 Introduction

In parallel to their large success, online social networks are also raising significant alarms. Users are beginning to question the mechanisms and policies that are used to protect their privacy and freedom of expression. The clamor about the PRISM program and the release of classified documents by Edward Snowden have sensitized a larger audience towards those issues of current social networking applications [15]. Thus, in many cases an approach based on peer-to-peer (P2P) or distributed technologies can be desirable. Attacks to distributed and P2P social platforms are yet possible. However, analysing these kinds of attacks is not the focus of the article. A comprehensive list of such attacks and countermeasures is presented in [12].

Instead, this article will deal with the possibility to use schemes for targeted broadcasting in a P2P social networking application. Some experiences have been acquired thanks to Blogracy, a new P2P system for social networking. The system is modular in the approach to the core problems of (*i*) data availability and resilience to censorship, (*ii*) content authenticability, (*iii*) data confidentiality, (*iv*) network anonymity, and (*v*) semantic interoperability. All these aspects are kept as much orthogonal as possible in the system. The system has been implemented and tested on the PlanetLab infrastructure. For both its architecture

O. Gaggi et al. (Eds.): GOODTECHS 2016, LNICST 195, pp. 203–212, 2017.
DOI: 10.1007/978-3-319-61949-1_22

and its level of implementation, to our knowledge it is quite unique. Blogracy is available as open source software (http://dev.blogracy.net/), thus it can be freely exploited for conducting further analysis and evaluations in the larger research area of distributed social platforms, exploring alternative architectural choices and implementations along each axis.

The rest of the article is organized in two main section: Sect. 2 presents background information about Attribute-Based Encryption (ABE) schemes and some related work about the realization of distributed social networking applications; then, Sect. 3 presents the security architecture of Blogracy and some experimental results obtained after its realization, in particular about the costs of network anonymization. Eventually, some concluding remarks are reported.

2 Background: ABE Schemes and Distributed Social Networks

In social networking and micro-blogging applications, it is often desirable to make some content available to a restricted audience, only. Access may be limited to the members of a circle of personal acquaintances, or authorized subscribers of a news channel. In some sense, this problem of confidentiality is similar to the case of broadcasting with DRM over an untrusted medium in general. Exploiting traditional public key cryptography and multicast group key management, it is possible to deliver a secret session key to intended recipients of confidential messages. This requires to rekey users periodically.

A recently emerging approach is to publish content, possibly on an insecure medium, in a form which can be decrypted only by users with proper attributes, as required by the content publisher's policy. In these Attribute-Based Encryption (ABE) schemes, in the general case the attribute authority is considered as a separate abstract entity. In fact, it may be an autonomous entity, and thus there is a third party ownership of the cryptosystem. Alternatively, the attribute authority may coincide with either the encryptor agent or the decryptor agent, creating a targeted broadcast [13], or a duty delegation scenario, respectively [20]. Those various alternatives are represented in Fig. 1. In particular, this work will focus on the targeted broadcast scenario (encryptor as owner).

2.1 Multi-authority ABE Schemes

The basic ABE schemes leave some open issues, especially with regard to the presence of many different Attribute Authorities (AAs). In fact, some practical use cases require that many different attributes can be defined in the network by different authorities, potentially corresponding to each single user. This regards the particular case of Blogracy, but also other generic P2P platforms for content distribution.

Chase [6] advanced one of the first proposals for a distributed CP-ABE scheme. In this scheme, all authorities are managed centrally by a trusted master authority. Attribute authorities can issue secret attribute keys autonomously.

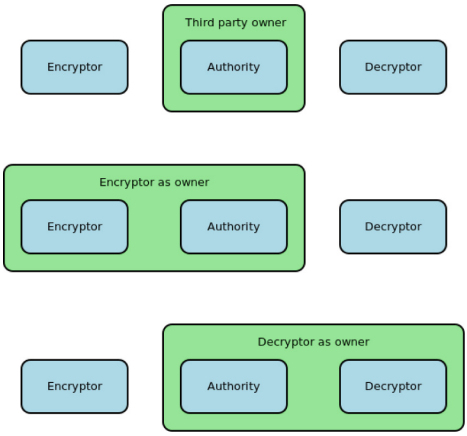

Fig. 1. Three scenarios for attribute authority ownership in ABE.

However, the addition of an attribute authority requires rekeying all users. A scheme proposed by Chase and Chow [7] removes the requirement for a central authority. However, it still relies on a pre-determined set of attribute authorities, which in fact need to coordinate during the setup phase for generating the distributed function. Müller et al. [21] propose another distributed extension of the CP-ABE scheme. Differently from Bethencourt et al. [4], they introduce algorithms for generating public and secret attribute keys, to be executed autonomously by some attribute authority. However, a central master authority is required to manage attribute authorities and users, to avoid possible collusion attacks. Lewko and Waters [17] describe a system where each node can act as an attribute authority, without any need for mutual trust or a master authority. An access policy can include attributes defined by different authorities. Secret attribute keys assigned to users still need to be distinguished, to avoid collusion threats. For this purpose, a hash of the users global identity is used. Unfortunately, this hash function is only modeled as an abstract random oracle, which is yet to be resolved to a concrete function. Wang et al. [24] describe a Hierarchical ABE (HABE) system. Basically, a root master authority empowers lower level attribute authorities, which can manage attributes and users at different levels, replicating the hierarchical structure of departments inside an organization. Wan et al. [23] propose a similar scheme named HASBE (Hierarchical Attribute Set Based Encryption). Lin et al. [19], instead, substitute the single central authority with a pre-defined set of attribute authorities, coordinating their operation during a setup phase. Anne et al. [2] propose a system which is quite similar to the previous one, using a secret sharing scheme. Table 1 summarizes the differences among the various proposals for Multi-Authority ABE.

Table 1. Features of various multi-authority ABE schemes.

System	Master authority (for managing AAs)	Pre-defined set of AAs (rekeying on changes)
Chase [6]	Yes	Yes
Chase and Chow [7] Lin et al. [19] Anne et al. [2]	No	Yes
Müller et al. [21] Wang et al. [24] Wan et al. [23]	Yes	No
Lewko and Waters [17]	No	No

2.2 Information Security in Distributed Social Networks

Various solutions are being proposed to overcome the centralized architecture of the most widespread social networking platforms. In particular, the following systems are designed according to a full-fledged P2P architecture. PeerSoN [5] is a system designed to provide encryption, decentralization and direct data exchange in the field of social networks, dealing with privacy and connectivity issues. Safebook [8] is based on a DHT and a network of socially close peers, defined Matryoshka. Peers in a user's Matryoshka are trusted and support the user by anonymizing communications and replicating content and profile information. Persona [3] is designed as a set of social networking services. For the confidentiality of users' profiles and data, it combines an ABE scheme with traditional cryptography, in such a way that a change in group membership requires widespread re-keying. LifeSocial [14] is a prototype developed over FreePastry for DHT indexing and PAST for data replication. Possibly, it is the more advanced system, with regards to implementation. For confidentiality, it uses a quite traditional encryption scheme. DiDuSoNet [16] is a multi-layered system, including (*i*) a lookup service based on a DHT and (*ii*) a Dunbar-based social overlay dealing with communication and storage of users' profiles. It exploits Dunbar's concept of tie-strength, thus reling on friend nodes with the highest level of intimacy [1].

Blogracy shares some architectural choices with the other distributed social networking systems that have been discussed above. It differs mainly in the adoption of an open and simple key-based identity system, and the use of interoperable protocols and widespread technologies, including OpenSocial, BitTorrent and Kademlia. It can be considered the first concrete attempt to cross the border between the application domains of file-sharing and social networking. Currently, it is one of the few systems which is freely available as a working prototype, as an open source project. Thus, it may represent an important resource for further analysis and testing in the whole research area of distributed social networks.

3 The Security Architecture of Blogracy

As discussed in the previous section, ABE schemes may be applied for targeted broadcasting (Fig. 1). In principle, they may also be useful in P2P systems, but some constraints remain about the need of a master authority, or some coordination mechanism among a predefined set of attribute authorities. The original ABE scheme has been adapted for use in Blogracy (http://www.blogracy.net/), a prototype P2P social network.

The architecture of the application is modular and is built around two basic components: (*i*) an underlying BitTorrent module for basic file sharing and DHT operations, exploiting an existing implementation, and (*ii*) an OpenSocial container, i.e., a module providing the services of the social platform to the local user, to be accessed through a web interface. For its basic operation, Blogracy exploits a P2P file-sharing mechanism and two logically separated DHTs [11]. Users in Blogracy have a profile and a semantically meaningful activity stream, which contains their actions in the system (e.g., add a post, tag a picture, comment a video). The first DHT (DHT1) associates the user's identifier with a reference to his social data. The user's social data is represented in a standard format, conforming to OpenSocial and ActivityStreams specifications (http://www.w3.org/wiki/Socialwg). It is encoded as a JSON file which contains all the elements described in OpenSocial Social API, including people, groups, activity streams, app data, albums, media items and messages, regarding a single user. All contained references, in the form of magnet-URIs, are keys of the second DHT (DHT2), which orchestrates the sharing of actual files, according to the BitTorrent protocol. In the following paragraphs, the security features realized in Blogracy over this extensible architecture will be described.

3.1 Identity and Authentication

Anonymity or pseudonymity are often a requirement of users of micro-blogging and other Internet applications. Nevertheless, user content needs to be verified for authenticity and integrity, properties which can be easily enforced by means of public-key cryptography and digital signatures. Usually, a public key is associated to a person or a legal entity through a certificate issued by a globally acknowledged authority. Instead, in Blogracy a user is represented directly though his own, locally generated, public key, according to a key-based identity scheme [18,22]. Thus, a user can reach other users only after obtaining their ID, i.e., the hash of their public key.

A signature scheme is used for attesting the authenticity of DHT entries, in particular the magnet-URIs used as values in the DHT1. The scheme is based on the JWS (JSON Web signature), which is currently an Internet Draft (http://tools.ietf.org/html/rfc7515). In fact, JWS is a quite practical specification, based on simple JSON objects and Base64 encoded strings. In particular, the header is a JSON object which specifies the cryptographic algorithms to use; the payload contains the signed message, which may be a JSON object or any byte sequence. Both the header and payload are encoded as Base64 strings, which are

concatenated (separated though a dot). A signature is then calculated, encoded as a Base64 string, and concatenated to the previously obtained text. At the end of the process, the JWS is represented as a concatenation of the three Base64 strings (header, payload, signature), separated by two dots. The header JSON object in use by Blogracy currently refers to the quite usual SHA256withRSA algorithm, named HS256 in the specifications. The "kid" field is an identifier for the signing key: in Blogracy the user's main public key is numbered as 0. The hash of this public key is also the user's unique identifier. Additional signing keys, corresponding to different kid values, may be added in the user's profile (which is included in the file containing all the user's social data, as an OpenSocial Person object).

The payload, instead, is simply the magnet-URI for the updated user's social data file. Verifying a DHT entry is handy, since the signing key corresponding to a certain kid can be found easily into the user's profile and then it can be stored locally by the user's followers for faster access (each signing key must itself be signed by the user's main public key). This way, the authenticity of an entry can be verified and the typical pollution of keyword-based DHT indexes can be easily detected.

3.2 Attribute Authorities

For its privacy model, Blogracy adopts an Attribute-Based Encryption scheme. It uses attribute credentials for protecting access to sensible content, creating a sort of very flexible personal circles of contacts, i.e., parametrized roles to be assigned to users for granting a certain set of access rights. The encryption scheme is based on the CP-ABE protocol [4]. In the current security model of Blogracy, each user is considered an authority for his own groups of contacts, in a typical Targeted Broadcast scheme. Each user can thus grant attributes and authorizations to his own contacts, and attributes are intended as defined in a namespace local to that user, without possible overlappings. All currently available multi-authority schemes have been discarded, because they do not fit a completely distributed P2P context. In fact, according to the design of Blogracy, it is not possible to identify a globally trusted central authority, even if distributed over a limited number of entities.

Nevertheless, Blogracy permits operation in a multi-authority scenario, though indirectly. In fact, in Blogracy attributes are attested by means of signed certificates (or delegation chains), originated by locally trusted attribute authorities. This way, other authorities can be taken into account, by allowing a user to show some attribute certificates. If the user proves to be eligible and the remote authority is trusted for those attributes, the user is issued a local attribute key.

Since each user is also an attribute authority for accessing his own social activities, then all relevant information is stored directly into his own JSON file, containing all the user's social data. In particular, in the "people" section, a list of acknowledged users can be created, as OpenSocial Person objects, where each user can be associated with a particular private attribute key. Each private attribute key is generated by the local user, using his own local master key MK

and a set of attributes S to associate with the acknowledged user. To keep the generated attribute key confidential, it is encrypted used the public key of the remote user. Additionally, the social data file of a user also contains his own public parameters (the PK data, according to the ABE scheme). Currently, encrypted data and plaintext data (SK and PK, respectively, according the ABE scheme) are stored in the appData section of OpenSocial Person objects.

Assigned attributes serve to mark group membership, and thus some social activities may be disclosed only to a certain set of groups. This is obtained by encrypting sensible social data, according to the desired policy. In Blogracy, an access policy for a resource is essentially a list of groups which are authorized for access. In fact, each object in the social data file can be encrypted according to a different policy; otherwise, it is kept as a plaintext JSON object.

3.3 Network Anonymity

Blogracy doesn't require users to expose their real identity and offers instead a pseudonimity mechanism based on public keys. However, anonymity is an issue also at the lower network level. In fact, if communications among users are based on direct connections, or file locations are expressed as plain network addresses, these can be easily associated with a particular person or entity. Various network technologies are being developed, which promise to guarantee a certain level of anonymity for their users, disguising their real network location. One of the best known is Tor (http://www.torproject.org/), but other similar networks are also available, which in fact may be more adequate for an application based on file sharing techniques.

In particular, I2P (http://geti2p.net/) is a anonymizing P2P overlay network, implementing a protocol similar to Tor. Since the I2P architecture is more distributed and focused on darknet-type services, it welcomes file sharing applications running inside the network (and thus separated from the larger communities operating on the plain Internet). Instead, Tor discourages the use of file sharing applications for not overloading its own outproxies. The operation of Blogracy over I2P requires the creation of a number of tunnels at each participant (router), determining their length, i.e., how many hops each tunnel should use. Thus security and latency can be balanced according to the needs of a particular application. Moreover, when used over I2P, BitTorrent clients do not support any Distributed Database, so they require a tracker to operate over I2P.

3.4 Experimental Results

In order to validate the feasibility of our approach, both simulations and performance tests were performed, as described in [11]. The evaluation confirmed our confidence about the realizability of a completely distributed social network, in the case of users with a sufficient number of followers and few relatively stable nodes. About the problem of data availability against the typical P2P node churn, we have discussed the results of some simulations in [11]. I particular, the

average notification delays are quite severe in the case of few followers, without stable nodes. However, when a user has more than 100–150 followers, the delays are negligible with just 5–10% of stable nodes. Thus, some aid has to be provided to newly joined users, in the form of resource hosting by other users, in addition to the small group of followers.

For testing the actual functioning of the system, twelve instances of Blogracy were executed over different remote nodes of PlanetLab Europe. For simplicity and consistency, the nodes were arranged in a small fully connected social network (i.e., each node follows all the others). Over this testbed, the pushing mechanism was verified. It is a direct form of communication, which nodes use most often. It allows a source node to notify its followers in a timely and effective way about the availability of update in the local social data file. Since it occurs directly between interested nodes (those participating in a swarm) it is virtually instantaneous [11].

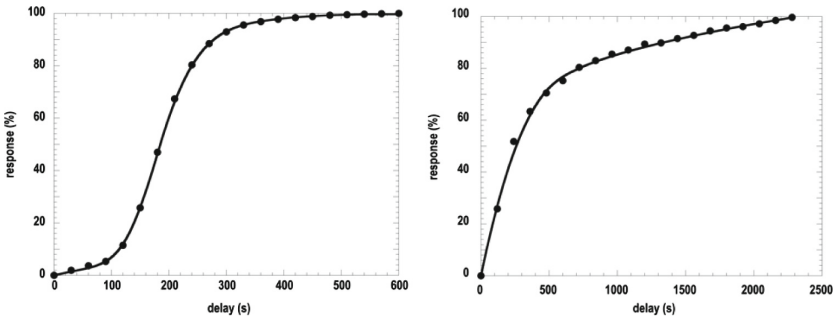

Fig. 2. Cumulative distribution of the notification delay in the reception of messages over the PlanetLab testbed; the polling cycle, repeated every two minutes, accounts on average for one minute. The first graph shows the case of normal Internet communications. The second graph shows the case of communications over I2P.

The polling mechanism was also verified. It involves sending various queries overs the DHT, downloading a torrent file and finally passing through download queues, before receiving the latest messages. It is worth noting that in a real system, this mechanism would be used only at the node startup, for receiving interesting activities produced when it was offline. Afterwards, push notifications would be more effective. In the tests, the polling cycle was set at two minutes, i.e., the availability of new messages from a certain source node was checked every two minutes. Probably, the more interesting result is the delay in the reception of new messages at follower nodes. The delays shown in the first graph of Fig. 2 are those concretely measured. In this setting, 90% of messages are received in 4.5 min. However, the measured times include a delay due to the polling cycle, which on average accounts for 1 min. Thus, in the mean delay of 4.5 min, 3.5 min are effectively due to DHT and BitTorrent download mechanisms. Analogous tests were finally conducted to evaluate the effectiveness of actually running

a distributed social network on top of I2P. In this case, an additional node was configured to host a tracker service on I2P. The cumulative distribution of measured delays is shown in the second graph of Fig. 2. The reception of 90% of messages requires around 20 min. With respect to direct connections, delays increase roughly by a factor of 4. It is possible that results could be improved slightly through finer setup, reducing the length of tunnels (and thus the level of anonymity). Results are in accordance with more extensive performance tests conducted over I2P [9].

4 Conclusion

This research work was aimed at studying a possible application of well-known P2P technologies, such as DHTs and BitTorrent, in the new domain of distributed social networking. In fact, although the primitives offered by those technologies were created with other goals in mind, however, they could be effectively adapted for Blogracy, a novel P2P micro-blogging and social networking system. Its main features are: (*i*) data availability and resilience to censorship, (*ii*) content authenticability, (*iii*) data confidentiality, (*iv*) network anonymity, and (*iv*) semantic interoperability. In particular, the security architecture of Blogracy has been presented, including its key-based identity system, a scheme for attribute-based content encryption with multiple authorities, and the option for network anonymization over I2P. In fact, the main aspects are kept as much orthogonal as possible in the system. Thus, it can also serve as a testbed for conducting further analysis and evaluations in the larger research area of distributed social platform, exploring alternative architectural choices and implementations along each axis. Moreover, some simulation results for notification delays and some empirical results obtained in test operations over PlanetLab have been presented. In particular, a quantitative comparison of plain and I2P anonymized communications has demonstrated that the latter implies much slower operations.

References

1. Amft, T., Guidi, B., Graffi, K., Ricci, L.: Frodo: friendly routing over dunbar-based overlays. In: 2015 IEEE 40th Conference on Local Computer Networks (LCN), pp. 356–364. IEEE (2015)
2. Anne, V.K., Praveen, G., Ramesh, N., Kurra, R.R.: Extension of MA-ABE for better data security in cloud. Int. J. Comput. Sci. Technol. (IJCST) **3**(1–1), A-419 (2012)
3. Baden, R., Bender, A., Spring, N., Bhattacharjee, B., Starin, D.: Persona: an online social network with user-defined privacy. ACM SIGCOMM Comput. Commun. Rev. **39**(4), 135–146 (2009)
4. Bethencourt, J., Sahai, A., Waters, B.: Ciphertext-policy attribute-based encryption. In: 2007 IEEE Symposium on Security and Privacy (SP 2007), pp. 321–334. IEEE (2007)
5. Buchegger, S., Schiöberg, D., Vu, L., Datta, A.: Peerson: P2P social networking: early experiences and insights. In: Proceedings of the Second ACM EuroSys Workshop on Social Network Systems, pp. 46–52. ACM (2009)

6. Chase, M.: Multi-authority attribute based encryption. In: Vadhan, S.P. (ed.) TCC 2007. LNCS, vol. 4392, pp. 515–534. Springer, Heidelberg (2007). doi:10.1007/978-3-540-70936-7_28
7. Chase, M., Chow, S.: Improving privacy and security in multi-authority attribute-based encryption. In: Proceedings of the 16th ACM Conference on Computer and Communications Security, pp. 121–130. ACM (2009)
8. Cutillo, L., Molva, R., Strufe, T.: Safebook: a privacy-preserving online social network leveraging on real-life trust. IEEE Commun. Mag. 95 (2009)
9. Ehlert, M.: I2P usability vs. tor usability a bandwidth and latency comparison. In: Seminar Report, Humboldt University of Berlin (2011)
10. Falkner, J., Piatek, M., John, J.P., Krishnamurthy, A., Anderson, T.: Profiling a million user DHT. In: Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement, pp. 129–134. ACM (2007)
11. Franchi, E., Poggi, A., Tomaiuolo, M.: Blogracy: a peer-to-peer social network. Int. J. Distrib. Syst. Technol. (IJDST) **7**(2), 37–56 (2016)
12. Franchi, E., Tomaiuolo, M.: Distributed social platforms for confidentiality and resilience. In: Social Network Engineering for Secure Web Data and Services, p. 114 (2013)
13. Goyal, V., Pandey, O., Sahai, A., Waters, B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Proceedings of the 13th ACM Conference on Computer and Communications Security, pp. 89–98. ACM (2006)
14. Graffi, K., Groß, C., Mukherjee, P., Kovacevic, A., Steinmetz, R.: Lifesocial.kom: a P2P-based platform for secure online social networks. In: 2010 IEEE Tenth International Conference on Peer-to-Peer Computing (P2P), pp. 1–2. IEEE (2010)
15. Greene, M.: Where has privacy gone? How surveillance programs threaten expectations of privacy. J. Marshall J. Info. Tech. Privacy L. **30**, 795 (2014). John Marshall J. Inf. Technol. Privacy Law **30**(4), 5 (2014)
16. Guidi, B., Amft, T., De Salve, A., Graffi, K., Ricci, L.: Didusonet: a P2P architecture for distributed dunbar-based social networks. Peer-to-Peer Netw. Appl. 1–18 (2015)
17. Lewko, A., Waters, B.: Decentralizing attribute-based encryption. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 568–588. Springer, Heidelberg (2011). doi:10.1007/978-3-642-20465-4_31
18. Li, N.: Local names in SPKI/SDSI. In: Proceedings of 13th IEEE Computer Security Foundations Workshop, CSFW-13. IEEE Computer Society (2000)
19. Lin, H., Cao, Z., Liang, X., Shao, J.: Secure threshold multi authority attribute based encryption without a central authority. Inf. Sci. **180**(13), 2618–2632 (2010)
20. Muijnck-Hughes, J.: Json Web Signature (JWS). Radboud University Nijmegen, Nijmegen (2011)
21. Müller, S., Katzenbeisser, S., Eckert, C.: On multi-authority ciphertext-policy attribute-based encryption. Bull. Korean Math. Soc. **46**(4), 803–819 (2009)
22. Tomaiuolo, M.: Trust management and delegation for the administration of web services. In: Organizational, Legal, and Technological Dimensions of Information System Administration, pp. 18–37 (2014)
23. Wan, Z., Liu, J., Deng, R.H.: Hasbe: a hierarchical attribute-based solution for flexible and scalable access control in cloud computing. IEEE Trans. Inf. Forensics Secur. **7**(2), 743–754 (2012)
24. Wang, G., Liu, Q., Wu, J., Guo, M.: Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers. Comput. Secur. **30**(5), 320–331 (2011)