From Access Control to Trust Management, and Back – A Petition

Dieter Gollmann

From Access Control to Trust Management, and Back – a Petition

Dieter Gollmann

Hamburg University of Technology,
Hamburg, Germany
diego@tu-harburg.de

Abstract. In security too often services are understood not from first principles but via characteristic mechanisms used for their delivery. Access control had got tied up with DAC, MAC, RBAC and reference monitors. With developments in distributed systems security and with the opening of the Internet for commercial use new classes of access control mechanisms became relevant that did not fit into the established mold. Trust Management was coined as a term unifying the discussion of those mechanisms. We view *trust* as a placeholder that had its use in driving this research agenda, but argue that trust is so overloaded that it is now an impediment for further progress. Our petition asks for a return to *access control* and proposes a new framework for structuring investigations in this area.

Denn eben wo Begriffe fehlen,
da stellt ein Wort zur rechten Zeit sich ein.
Mit Worten läßt sich trefflich streiten,
mit Worten ein System bereiten.

[Mephistopheles in Goethe's Faust 1.]

1 Services and Mechanisms

From communications security we get the important conceptual distinction between *security services* and *security mechanisms* [7]. A service describes security goals that should be achieved at a generic, implementation independent level. A mechanism is an implementation of a security service. Implementations may reflect specific requirements of a class of applications or specific features of a technology.

This distinction is useful, although sometimes difficult to maintain in practice when services become equated with the characteristic (sic!) mechanisms used for their delivery. This has led to definitions such as "authentication is what authentication protocols do". There is a further problem. In case a service is too closely tied to its 'old' mechanisms and the applications or the technology changes, the service may get renamed just to break clear of its past implementations although the security goals have actually not changed.

In an early attempt to define the yet nascent field of informatics, Zemanek followed Mephistopheles from Goethe's Faust when making the salient point that we need words for discussing emerging phenomena *before* they are truly understood [17]. The terms coined at that stage are placeholders, yet without precise meaning, just vehicles for moving the discussion along. Zemanek concluded that the new word 'informatics' was such a placeholder. In security, a similar pattern can be observed when old mechanisms become insufficient and have to be substituted by something new, but it is not yet fully determined what the new solution should provide.

2 Access Control

Access control is one of the security services listed in [7].

> *Access control*: provides protection against unauthorised use of resources.

From the 1970s on, the corresponding security mechanism was a *reference monitor* enforcing security policies that referred to user identities or to security labels. The former policies were called discretionary access control (DAC), the latter mandatory access control (MAC) [1]. The reference monitor was implemented by the *security kernel*. On a historic note, *trusted* operating systems in the 1980s were those that supported multi-level security based on security labels.

By the mid 1990s access control by DAC and MAC, which had deep roots in the US defense sector, showed itself unsuitable for the commercial applications that came to dominate the use of IT. In role-based access control (RBAC) security policies refer to functional roles in an organisation, not to user identities or to security labels [14].

The Internet had been opened to commercial use in the early 1990s creating opportunities for Internet-based interactions between organisations. Before, access control was a purely local service, both with respect to the setting of policies and with respect to their enforcement. Exposure to the Internet led to demands for new kinds of access control. In particular, there were fundamental changes in the enforcement of security policies.

3 Trust Management

In 1996 Blaze *et al.* introduced the term *Trust Management* for access control in this new environment [3]. As stated in [2]:

> Trust management, introduced in the PolicyMaker system, is a unified approach to specifying and interpreting security policies, credentials, and relationships.

As a service, Trust Management is nothing else but access control[1]. As a mechanism, it is a distinctive departure from the past.

[1] The purist may complain that this service is defined by reference to a collection of mechanisms.

As a service, access control is a system for describing and interpreting policies that regulate access to resources. Policies refer to *attributes*. There is no inherent limitation on the attributes that can be used for access control. User identities and security labels were just two instances of convenient attributes. Attribute values (evidence) should not be taken on trust and have to be authenticated.

> *Authentication*: decides whether to accept or reject claimed evidence provided with an access request.

In the past the main policy attribute was the user identity. From that time, authentication can be narrowly understood as the verification of a claimed user identity. In the past, evidence was verified locally. Now, verification might be 'delegated' to some other entity which then advises on the validity of an attribute value. Attributes thus have a value and a source. Authentication is split into the verification of the attribute value, possibly performed externally, and the local verification of the source of the advice received from an external entity. The SAML specification refers to trust when discussing the use of XML Signatures for origin authentication [13].

Authentication was once reserved for origin authentication, as in *obtaining the source of the request is called 'authentication'* [12]. Now, a new term is needed for the verification of an attribute value. In code-based access control, for example, *code-identity authentication* goes beyond verifying the source of code [11].

Access control also includes the step where for a given request a decision is made based on the current policy and on the evidence presented. KeyNote calls this step *compliance checking* [2]. In the past, this step was called *authorisation*. As put famously in [12],

$$\text{access control} = \text{authentication} + \text{authorisation}.$$

In common use of English people are authorised but requests (transactions) are approved. Hence, we may put instead

$$\text{access control} = \text{authentication} + \text{approval}.$$

Figure 1 captures the new view of access control. A *Policy Decision Point* (PDP) receives a request together with a set of attributes. The request may arrive within a *session*. For externally verified evidence, the source of evidence has to be authenticated. Locally verifiable evidence is directly authenticated. The session identifier may associate the request with further local evidence, *e. g.* with the identity of an authenticated user who had established the session. Once authentication is completed, the PDP decides whether to approve the request.

4 Trust and Authorisation

Descriptions of access control mechanisms explain how a request will be approved in accordance with the given policy. Such explanations state how attributes are authenticated, how the applicable policy rules are found, and how those rules are to be interpreted. Such descriptions do not explain how the policy

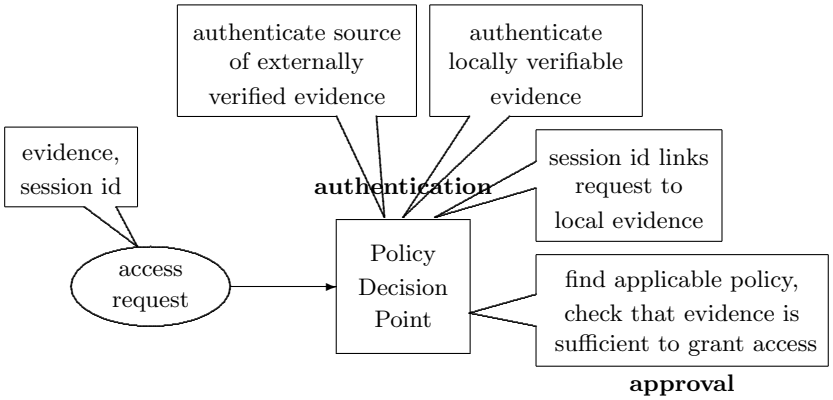

Fig. 1. A fresh look at access control.

came into existence in the first place. The act of defining a policy can be called *authorisation*. A policy based on user identities can authorize a user to perform certain actions; a policy based on roles can do the same for the functional roles in an organisation. The SAML specification refers once more to trust when discussing authorisation.

> When determining what issuers to trust, particularly in cases where the assertions will be used as inputs to authentication or authorization [approval in the terminology proposed above] decisions, the risk of security compromises arising from the consumption of false but validly issued assertions is a large one [13].

An organisation has some rationale for setting its policy. A good example for such a rationale is the *need-to-know* principle; users and roles must be enabled to perform the actions expected from them. A policy is set by assigning access rights (permissions) to *principals* such as users, roles, code, web sites, *etc.* When 'trust' is used to capture the scope of access rights granted, *e. g.* as in *code with more trust is allowed to do more on your machine* [11], language suggests that access rights are granted *because* an entity is trusted in an anthropomorphic meaning of the word.

It is plausible that trust in a person or in an institution and the access rights granted are correlated. In the social fabric of an organisation decision makers will appoint people they trust to positions of responsibility. Equally, appointment of contractors will relate to trust established earlier. We are, however, departing from access control when speculating about the reasons driving decisions within an organisation. In this context, a mixture of competence, reliability, cost, and personal allegiance will play its role. There exists advice for organisations on how to best balance these various factors in their decisions, and one should note that when too much emphasis is placed on personal allegiance and family ties, trust just becomes another word for nepotism.

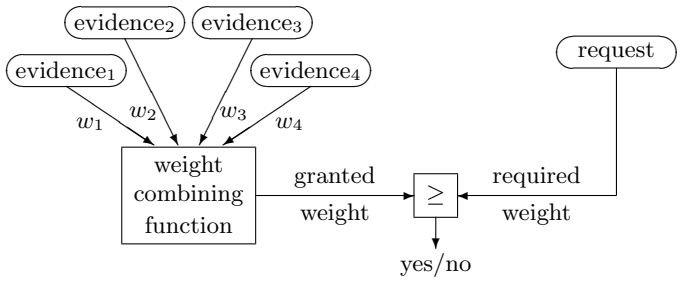

Fig. 2. Weighing evidence for access control.

5 Weighing Evidence

It is desirable that a security policy should be *complete*; for each request and for each viable combination of attribute values there should be a defined decision. Moreover, all the policy rules ought in principle be accredited by management. In a setting with a rich collection of attributes dealing with each possible case individually becomes unmanageable. An alternative approach discussed in [9] structures access control as follows.

- Each piece of evidence gets assigned a granted weight; the weight may reflect the trustworthiness of evidence.
- An algorithm for combining the weights of different pieces of evidence is defined; in the simple most case, weights are just added up.
- Each request gets assigned a required weight.
- A request is approved if the combined granted weight of the evidence presented exceeds the required threshold.

The weights assigned to evidence might be called trust but this term evokes an interpretation that is not necessary. Weight of evidence is an established concept, as are decision processes that require judge or jury to move beyond reasonable doubt when finding their verdict. Practical validation of the approach to access control sketched here still is an open challenge.

6 Reputation as an Attack Vector

It has been proposed to use *reputation* (a.k.a. trust) as an attribute for access control, see *e. g.* [4, 5, 10, 15, 16]. This approach is problematic for two reasons. First, we have to deal with the authentication of this attribute. When reputation scores are computed locally we can believe what we see but still should account for imperfections in our observations. When reputation scores are received as *recommendations* from third parties, we need a policy for accepting external evidence. We may take the ratings from certain entities on trust. Alternatively, we could iterate the process and rate a party's reputation as a recommender; we

could, for example, compare recommendations received with our own observations and trust recommenders that share our bias.

There is a second problem. The use of reputation inherently assumes that the past is a good predictor for future behaviour. Note that prediction need not be deterministic and can very well incorporate probabilistic reasoning. This places us in the domain of reliability; we get meaningful guarantees as long as we have a reasonable statistical model for an entity's behaviour. By the same measure, we do not have a security mechanism. Intentional attacks do not follow established statistical patterns.

To the contrary, when we rely on reputation for access control we open ourselves to attacks known as *confidence fraud*. The attacker follows a course of action that leads to a good reputation score and then strikes at a convenient moment. eBay is often quoted as an example for the successful deployment of reputation systems, usually without giving a definition for success. In reality, eBay's reputation system could be gamed and was, *e. g.*, modified in 2008 to deal with some of the more blatant misuses[2].

If reputation-based access control is to be taken seriously as a security mechanism, it has to be subjected to proper security analysis. Reputation-based systems are usually assessed via simulations, und usually the adversaries deviate at random from the correct behaviour. Such an approach is suitable for a reliability analysis. It is fundamentally flawed in the context of security. A security analysis deserving its name is a min-max method that first looks at the maximal damage an attacker can cause for a specific defence (within given assumptions on attack patterns of interest) and then searches for the defence that minimizes the maximal damage.

7 Occam's Razor

Our discussion of access control has encountered seven forms of trust:

- Trust as an indicator for multi-level security and mandatory access control.
- Trust as a synonym for access rights, such as in trusted code or in semi-trusted code.
- Trust as origin authentication (trust in an assertion)
- Trust as a policy rule to accept evidence from a third party (trust in an issuer/recommender, delegation of authentication).
- Trust as a rationale for assigning access rights.
- Trust as a weight of evidence.
- Trust as a synonym for reputation.

Occam's razor has been expressed as

> *pluralitas non est ponenda sine necessitate*
> (plurality should not be posited without necessity).

There is no necessity to use a term like trust that has such a plurality of meanings, a point that had been elaborated in more detail in [8].

[2] `http://www.wired.com/epicenter/2008/05/ebay-feedback/`

8 The Petition

Influential work on access control was published in the 1970s and 1980s. Commercial use of the Internet then changed the context for access control and it became apparent that the old security mechanism had reached their limits. Trust management was forged as a new term. As put by Joan Feigenbaum, one of its creators:

> Trust management is supposed to be an incredibly vague and provocative term invented by Matt Blaze. I don't know whether he intended it that way, but it comes natural to him [6].

The petition is then to return to *access control* as this term quite adequately captures the very nature of this security service. Challenges in access control lie in the identification of suitable policies and policy attributes, and in the authentication of those attributes, set in the context of federated (mashed-up) systems where access control functions are distributed among many players. All of this can be expressed quite elegantly without mentioning trust. Trust has served its purpose as a placeholder and catalyst in the discussion transforming access control and can be safely put to rest.

References

1. D. E. Bell and L. J. LaPadula. Secure computer systems: Mathematical foundations and model. Technical Report M74-244, The MITRE Corporation, Bedford, MA, May 1973.
2. Matt Blaze, Joan Feigenbaum, John Ioannidis, and Angelos D. Keromytis. The KeyNote Trust-Management System Version 2, September 1999. RFC 2704.
3. Matt Blaze, Joan Feigenbaum, and Jack Lacy. Decentralized trust management. In *Proceedings of the 1996 IEEE Symposium on Security and Privacy*, pages 164–173, 1996.
4. Piero A. Bonatti, Claudiu Duma, Daniel Olmedilla, and Nahid Shahmehri. An integration of reputation-based and policy-based trust management. In *Semantic Web Policy Workshop in conjunction with 4th International Semantic Web Conference*, Galway, Ireland, November 2005.
5. Maurizio Colombo, Fabio Martinelli, Paolo Mori, Marinella Petrocchi, and Anna Vaccarelli. Fine grained access control with trust and reputation management for globus. In *Proceedings of the 2007 OTM confederated international conference on On the move to meaningful internet systems*, pages 1505–1515, 2007.
6. Joan Feigenbaum. Overview of the AT&T Labs trust-management project. In *Security Protocols, LNCS 1550*, pages 45–50. Springer Verlag, 1998.
7. International Organisation for Standardization. *Basic Reference Model for Open Systems Interconnection (OSI) Part 2: Security Architecture*. Genève, Switzerland, 1989.
8. Dieter Gollmann. Why trust is bad for security. *Electronic Notes on Theoretical Computer Science*, 157(3):3–9, 2006.

9. Audun Jøsang, Dieter Gollmann, and Richard Au. A method for access authorisation through delegation networks. In R. Safavi-Naini, C. Steketee, and W. Susilo, editors, *Proc. Fourth Australasian Information Security Workshop (Network Security) (AISW 2006)*, pages 165–174, Hobart, Australia, 2006. CRPIT, **54**.
10. Karl Krukow, Mogens Nielsen, and Vladimiro Sassone. A logical framework for history-based access control and reputation systems. *Journal of Computer Security*, 16(1):63–101, 2008.
11. Brian A. La Macchia, Sebastian Lange, Matthew Lyons, Rudi Martin, and Kevin T. Price. *.NET Framework Security*. Addison-Wesley Professional, Boston, MA, 2002.
12. Butler Lampson, Martín Abadi, Michael Burrows, and Edward Wobber. Authentication in distributed systems: Theory and practice. *ACM Transactions on Computer Systems*, 10(4):265–310, November 1992.
13. OASIS. Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) v2.0. Technical report, OASIS Standard, March 2005.
14. Ravi S. Sandhu, David Ferraiolo, and Richard Kuhn. The NIST model for role based access control: Toward a unified standard. In *Proceedings of the 5th ACM Workshop on Role Based Access Control*, pages 47–63, July 2000.
15. Sangheethaa Sukumaran and Elijah Blessing. Reputation based localized access control for mobile ad-hoc networks. In *Proceedings of Ad-Hoc, Mobile, and Wireless Networks, LNCS 4104*, pages 197–210, 2006.
16. Huang Yong. Reputation and role based access control model for multi-domain environments. In *2010 International Symposium on Intelligence Information Processing and Trusted Computing (IPTC)*, pages 597–600, October 2010.
17. Heinz Zemanek. Was ist Informatik? *Elektronische Rechenanlagen*, pages 157–161, 1971.