Dagstuhl Seminar 10141: Distributed Usage Control

Sandro Etalle[1], Alexander Pretschner[2], Ravi Sandhu[3], Marianne Winslett[4]

[1] University of Twente, The Netherlands
[2] Karlsruhe Institute of Technology, Germany
[3] University of Texas at San Antonio, USA
[4] University of Illinois, USA

Abstract. We summarize Dagstuhl seminar 10141 on Distributed Usage Control.

Keywords: Usage control, access control, data protection, privacy, security policies, trust, trusted computing, compliance, DRM, information flow

In general, access control defines who may access which data, and under which circumstances. A good access control system is at the base of every process which handles confidential information. As an extension to access control, usage control is about defining and enforcing how data may or may not be handled after it has been accessed (e.g., "do not disseminate," "delete after thirty days," "notify me when accessed," “use only for scientific purposes.”) Usage control is particularly relevant when it comes to privacy, protection of trade secrets or intellectual property, digital rights management, and auditing/compliance in the context of regulatory frameworks. Usage control, considered as a branch of information security, is hence both relevant for society and economics.

While there is a pressing need for usage control, existing solutions are partial – e.g., via access control mechanisms – and often specialized. The problem is particularly challenging in distributed environments where servers, which give away data, can neither see nor control what clients do with the data after their reception. In this setting, enforcement can be accomplished in one of two ways: by ensuring that policies are not violated, or by detecting and reporting violations, online or off-line. These two approaches apply in different technological environments, and they apply to different underlying trust and business models.

The technical problem of usage control spans at least the following four dimensions that we addressed in the seminar:

- Enforcement mechanisms and their guarantees: From a technical perspective, enforcement (either guaranteed or achieved by detecting policy infringements) is arguably the most challenging problem. Abstractly speaking, both of the above-mentioned kinds of enforcement mechanisms consist (1) of a condition in which the mechanism is supposed to perform its task – a monitoring component is hence needed – and (2) of an action. That

Dagstuhl Seminar Proceedings 10141
Distributed Usage Control
http://drops.dagstuhl.de/opus/volltexte/2010/2717

action can be an inhibition: the future "usage" of a data item is simply disallowed, i.e., its rendering, management, disclosure, or execution. The action can also be the execution of a command: action requirements, e.g. deletion or anonymization requirements for privacy reasons are satisfied by simply executing deletion or anonymization commands. Both the monitoring and the action part have to be implemented in a trustworthy and tamper-proof way; ideally, they are also transparent to the underlying system and hence can be applied to legacy systems. The enforcement can be done at many different levels, including the level of hardware, operating systems, virtual machines, runtime systems, services, and business processes.

- Information flow: Because non-dissemination arguably is the most relevant instance of a usage control requirement, usage control can in many cases be seen as an information flow problem: a data item, or parts or transformations of that data item are not allowed to be transferred to another subject. We consider the information flow problem primarily in dynamic settings. Experience shows that fully inhibiting information flow tends to restrict a system to a point where it cannot be used any more in any useful way. A fundamental and unresolved question then is that of allowing an "acceptable" amount of information flow.
- Policies: Usage control requirements must be specified in policies. While the temporal dimensions in this context seem to have been largely understood, the semantics of propositions remains unclear. For instance, what does it mean to "delete" a data item – does it mean to just delete a FAT entry, to overwrite file 20 times with random byte patterns, to delete all backup copies as well, not to allow any copying operations before? It appears that a definition of these semantics must relate to the different levels of enforcement as described above. Moreover, how do these policies evolve over time, that is, when data is disseminated, possibly after having been altered? A further fundamental problem here is the trade-off of usability and expressivity.
- Trust: Tightly bound to the problem of re-disclosing data is that of trust. Some subjects may be entitled to receive data, but how do we know, and how do they prove it? How do we cope with the problem of evolving trust relationships?

Different communities from fields as diverse as trust management, databases, operating systems, information flow, access control, DRM, trusted computing, compliance, social networks, general security and service-oriented architectures, but also from application domains such as eGovernment and the health sector, have, largely independently, worked on concepts, formalisms, tools, systems, and guarantees that relate to distributed usage control. The goal of this Dagstuhl seminar was to bring together these different communities, identify gaps and overlaps, and foster collaboration between theory and practice.

With about 50 attendants, the Dagstuhl seminar on Distributed Usage Control has had an overwhelming response to the invitations that were sent out. One noteworthy characteristics of the seminar was its multidisciplinary nature. Security is not only technical; it is a multidisciplinary field that has legal, regulatory and societal aspects too. This makes security research particularly challenging. This Dagstuhl seminar had

a technical core, but sparked discussions also from neighboring fields, in particular a plethora of issues related to privacy. This gave rise to three days of lively discussion, with a regular interleaving of general agreements and disagreements.

One of the fixed points of the meeting was that securing data and applications is a tremendously difficult problem, which cannot be fully solved. This, however, should not withhold us from improving security techniques, which is something that can and has to be done. Quoting one of the attendants: “In the last ten years, people have been calling "security" what is really best-effort security or soft security or perhaps not security at all: they have weak attacker models or perhaps not enough analysis. But a system that is 99.9% secure is very different from a system that is 80% secure. We should not say that perfect security is impossible and then not just try to get close to it.’’ This led to strong disagreements as several other attendants emphasized the trade-off between security, usability, cost, and feasibility.

Indeed, the community is aware and has worked on a number of important problems, and solved in part or in total many of them. The field of security in general and of usage control in particular is rich of success stories, and of technical breakthroughs that have made computers much more secure. This however has certainly not solved the problem of security in general. On the contrary, we note a kind of law of conservation of misery affecting security (and access control and usage control) research. Quoting one of the attendants: “[for instance] you can add typing to Javascript, thereby shifting the blame for problems from the language, then to researcher, then the programmer, and then finally shift it to the end user.” We should stop trying to shift the blame.”Instead, we should assume that the problems are here to stay, and come up with some solution approaches.’’

An indication given during the conference was to look at imperfect security as being a relevant part of the field. As pointed out in the lecture of one of the attendants, the usual fundamental definitions concerning runtime monitoring consider only “successful” runs, i.e., runs in which none of the policies is infringed. The field is lacking the fundamental results on how to address infringements which, as an aside, seems to be true for many fields of computer science where exceptions have to be managed.

In any case, we have to take note of the fact that the security and privacy fields seem to be missing a unifying vision, a unique driving force behind all security research. Quoting one of the attendants: “We have some excellent papers […], but my gut feeling is that we don't know what the real issues are right now in security. What is driving this whole thing? We have type systems, language based security, and so on, but where is the equivalent of what PKIs did, in one or the other way, for internet commerce?”

We believe that this sort of intrinsic fallibility of security, and the lack of unifying vision just mentioned are due to the multidisciplinary aspect of the field. Perhaps because of this, it is still difficult to get security & privacy in the sight of legislators and policy makers in such a way that accountability is handled properly. One

attendant pointed this out and suggested to “write the conclusions of your papers such that politicians and the general public can understand them. If your conclusions are to make policies more detailed, then be careful to keep in mind the effect if those conclusions could appear in the media.”

In sum, the seminar enjoyed a somewhat unexpected focus on privacy-related issues and intense discussions on the general subject of security research and its connection or disconnection with real-world problems. To the surprise of some, there continues to be disagreement on whether 100% security is a desirable goal, even though it is unlikely to be reached, or if pragmatic considerations including cost, feasibility, usability, innovation and fun should rather lead to a risk-based approach that aims at imperfect security, and if the community shouldn’t strive to understand what the risks are, and what imperfect security really is.