Chapter 3.25
Security and Trust in P2P Systems

Michael Bursell
Cryptomathic, UK

ABSTRACT

This chapter examines the issue of security in peer-to-peer (P2P) systems from the standpoint of trust. It takes the view that P2P systems present particular challenges in terms of trust over other socio-technical systems, and identifies three key areas of importance: identity; social contexts; punishment and deterrence. It suggests that a better understanding of these areas and the trade-offs associated with them can help in the design, implementation, and running of P2P systems. The chapter combines a discussion of problems and issues in current systems with a review of some of the wider sociological and nonsystems literature which can aid those involved with P2P systems. It concludes with some suggestions for areas where future research may provide fruitful insights.

TRUST AND SECURITY

"I would trust my brother or my sister with my life, but I wouldn't trust either of them to back-up my hard drive."

Peer-to-peer (P2P) systems require different entities to decide how to interact with others—or whether to interact with them at all: these are security decisions. The system itself will probably be set up to allow particular types of interaction, or to allow particular choices about interaction: these, too, are security decisions. They are in fact decisions about trust. Within many P2P systems, I need to know whether I can "trust" another entity within that system, Alice, and what to do with a statement from yet another entity, Bob, saying that *I* can trust this Alice because *he* does. "Trust" is a word that is used very loosely in English, but a concept that should exercise the thoughts of anyone thinking about security in a computer system, particularly when that system

is distributed, and even more so when it is a P2P system. This chapter addresses how trust and security are linked in certain types of P2P systems and provides some ways of understanding how assumptions about trust can help or hinder the security of a system. It takes a sociological, rather than an overly technical view, with the hope that from such a perspective, designers, builders, and users of P2P systems may have a better chance of achieving the kinds and levels of security they need.

Why Does Trust Matter?

One approach to dealing with how to build trust into a system is that discussed by Waldman and Rubin (2001): "We are concerned less about conceptions of trustworthiness than we are about designing systems that rely as little as possible on trust. Ultimately, we would like to design systems that do not require anyone to trust any aspect of the system ... In this context the ideal trusted system is one that everyone has confidence in because they do not have to trust it. Where that is impossible we use techniques such as reputation building and risk reduction to build trust" (pp. 243–244). For them, trust is an issue to be avoided if possible, and built in if absolutely required. This chapter argues that trust is a key component of P2P systems, whether implicit or explicit, and that understanding the requirements, costs, and trade-offs around it is vital to such systems' design, implementation, and running.

It is worth stating at the outset that this chapter attempts to examine a broad superset of P2P systems and that for some subsets some or many of the lessons addressed may not be relevant. The "perfect" P2P system could be defined as a system with a perfectly flat hierarchy, full communication between all entities, and with each entity allowing other entities to use local resources (storage or processes) according to some agreement or contract, whilst simultaneously using remote resources provided by other entities. While this kind of system may never exist, the aim of this chapter is to look at the kinds of security questions that arise in such a world: other systems can hopefully be examined as subsets or refinements of such a "perfect" P2P system.

One of the great difficulties in thinking about security for P2P systems is that the model of security that must be espoused is different from that of most other types of system. Most other systems have a centralised entity "providing security," though that centralised provision may well be delegated. Without a centralised entity "providing security," requirements and assumptions about security in the system need to be built not only into the fabric of the system (its infrastructure) but also into the models of interaction between entities—this is one of the challenges facing us in the P2P world. In most other systems, there is an assumption that attackers, in the sense of entities who perform "intentional and unwarranted actions" (Schneier, 2003), are most likely to come from without, or at least that internal attackers can be profiled and controlled. In many P2P systems the danger is reversed in that attackers can come from within the system just as easily as from without. The challenge of mitigating the actions of attackers must be tackled at the design stage of system, as it if for the system itself, rather than a godlike administrator (or her minions), to deal with such attacks. If a new way of interacting between entities in the system arises that has not previously been considered, there can be no magic fix that can be applied. A good example of this is the story (later discredited) that the Recording Industry Association of America (RIAA) was planning to "infect MP3 files in order to audit and eventually disable file swapping" (Orlowski, 2003). The P2P systems that were distributing MP3s had no model to deal with malicious insiders distributing malware content: the (implicit) assumption was that all content could be "trusted."