The Trust Economy of Brief Encounters

Ross Anderson
University of Cambridge Computer Laboratory
Ross.Anderson@cl.cam.ac.uk

February 14th, 2009

The security of ad-hoc networks has been a subject of academic study for about fifteen years. In the process it's thrown up all sorts of provocative ideas, from Berkeley's Smart Dust to our suicide bombing protocol [1, 2]. Now that a number of ad-hoc network technologies are being deployed, it turns out that reality is even stranger. After giving an overview of the history, I'll look at what some real systems teach us.

Peer-to-peer systems were our first contact with this world; the ambitiously-named Eternity Service may be no more, but it inspired a host of other peer-to-peer systems that brought creative mayhem to the music industry and helped teach us the importance of incentives; people are more likely to defend things they care about than the more abstract free speech of remote users on unfamiliar topics [3, 4]. More broadly, a literature grew up on reputation systems: the key idea here was that the initial establishment of a secure association was much less important than mechanisms to deal with its evolution afterwards. No-one can remember when they first decided to trust their mother, and many of us can't remember when we first met many of our friends [5]. Trust is organic; it grows and can also die.

The Resurrecting Duckling was our second contact [6]. Here the idea was to bootstrap trust using physical contact, as a means of introducing practical authentication into embedded systems for which the overhead of PKIs and crypto protocols was unacceptable. It found application, for example, in digital tachographs: the tachograph sensor acts as the mother duck and the vehicle unit, on initialisation, imprints on it by accepting a key offered by the sensor [7]. That this key is sent in the clear is immaterial. The threat model is that the truck driver, perhaps months after the official calibration of the device, attempts to tamper with the communication between the sensor and the vehicle unit. Wiretapping of the key set-up is excluded by the environmental assumptions.

A third development was the HomePlug protocol [8]. This is used to support power-line communications between consumer electronic devices, and comes with two modes of operation: "simple connect" mode, which works like a tachograph by sending the key in the clear, and "secure mode" in which the user types a key provided on the device label into a network management station. An academic cryptographer might have hoped to see a protocol such as Diffie-Hellman here. We decided against this because users would still have to either verify a key checksum or copy a key in order to prevent middleperson attacks, and from the viewpoint of security usability, copying is much preferable to checking. It's also cheaper and more robust to do things simply.

A fourth development has been recent work on social networks, which taught us about the importance of topology, and gave us new insights into such matters as traffic analysis and anonymity. Again, the big issue isn't the key setup per se, but how you manage the evolution of the peer relationships afterwards. Node compromise is a big deal in some of these systems, while in others it's group compromise – the detection by authority of a covert community [9].

The latest development comes from industrial control systems, where some vendors are selling what they call "lick'em and stick'em" sensors that can be deployed rapidly in an industrial plant. For example, a process engineer can add an extra temperature sensor to a reactor vessel by just slapping it on the outside and letting it establish its own network to the control centre. At first sight, that can save a lot of money on cabling. But configuration management now becomes the bugbear: Homer takes the sensor destined for the tank of methyl isocyanate and slaps it in the sewage tank instead. And that isn't all. Ad-hoc deployments into private networks open up backdoors to the Internet that bypass the plant firewall, and there's always the small matter of battery replacement. An initially appealing solution can rapidly teach some hard lessons about lifecycle costs.

In short, the things that caused problems mostly weren't the things we'd expected to. The protocols community's traditional insistence on immaculate key establishment turned out to be overblown or even irrelevant in most of these applications; what mattered was what happens afterwards. It's about putting the effort into the right part of the security lifecycle. But that effort is still substantial, and if anything it's much greater than the cost of initial authentication.

So what does this teach us about brief encounters? I will use our now traditional definition of trust as the ability to do harm – a trusted system or component is one that can break your security policy. From this I believe a difficult conclusion follows. If you are going to trust a principal with whom you have only one brief encounter, then the mechanisms commonly associated with "ad-hoc networks" – namely the optimistic establishment of an association, followed by its continued assessment on the basis of subsequent behaviour – are inappropriate. They really work only where the encounters are repeated or prolonged; only then

do tit-for-tat and various institutional and social sanction mechanisms kick in effectively.

Where the encounter is brief, the trust will usually have to be provided by some third party, most likely using one of the many conventional protocols discussed at these workshops in the past. A merchant won't give goods to a customer without either trousering her cash or executing an EMV protocol run with her chip card. Whether you trust the Bank of England to issue the notes or the Bank of Scotland to issue your chip card, you're still trusting a bank.

Of course this is widely misunderstood. Websites tiresomely demand passwords from one-off customers when all that really matters is the card transaction. Transactions are often asymmetric because of a power imbalance; the difficulty that bank customers have in telling fake PIN entry devices from real ones is just one example of many. Nonetheless a more mobile and fluid society, with more transient encounters, will on aggregate require more security protocol runs. Perhaps they'd be designed better if we were more explicit about two things: how long the resulting security associations are intended to last, and who's guaranteeing what to whom. Knowing someone's name isn't the same as knowing their reputation, and being able to tarnish their reputation is usually little recompense afterwards (ask any of Mr Madoff's erstwhile customers). Clarity about these questions might hopefully shift the emphasis in many applications from "identity", whatever that is, to something more appropriate.

References

[1] Joseph Kahn, Randy Katz, Kris Pister, "Emerging Challenges: Mobile Networking for Smart Dust", in *Journal of Communications and Networks* v 2 no 3 (2000) pp 188–196

[2] Tyler Moore, Jolyon Clulow, Shishir Nagaraja and Ross Anderson, "New Strategies for Revocation in Ad-Hoc Networks", in *ESAS 2007*, Springer LNCS 4572 pp 232–246

[3] Ross Anderson "The Eternity Service", in *Proceedings of Pragocrypt 96* pp 242–252

[4] George Danezis and Ross Anderson, "The Economics of Censorship Resistance", at WEIS 2004 and in *IEEE Security & Privacy* v 3 no 1 (2005) pp 45–50

[5] Ross Anderson, "The Initial Costs and Maintenance Costs of Protocols", in *Security Protocols Workshop* 2005 Springer LNCS v 4631 pp 333–343

[6] Frank Stajano, Ross Anderson, "The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks", in *Security Protocols, 7th Interna-*

tional Workshop (1999), http://www.cl.cam.ac.uk/~rja14/duckling.html

[7] Ross Anderson, "On the Security of Digital Tachographs", in *Computer Security – ESORICS 98*, Springer LNCS vol 1485 pp 111–125

[8] Richard Newman, Sherman Gavette, Larry Yonge and Ross Anderson, "HomePlug AV Security Mechanisms", in *ISPLC 2007* pp 366–371

[9] Ross Anderson and Shishir Nagaraja, "The Topology of Covert Conflict", at WEIS 2006; University of Cambridge Computer Laboratory technical report CL-637, 2005