

Authentication in Pervasive Computing: Position Paper

Irfan Zakiuddin¹, Sadie Creese¹
Bill Roscoe², and Michael Goldsmith³

The pervasive computing paradigm foresees communicating and computational devices embedded in all parts of our environment, from our physical selves, to our homes, our offices, our streets and so forth. What will security mean in this New World of ubiquitous computing? In this position paper we outline our current thinking on the new *issues and problems* in security for pervasive computing, the emphasis being on delineating the range of new technological issues and problems.

In this new paradigm devices will need to interact, almost spontaneously, with certain other devices in an environment that is both unknown and changing. In traditional approaches, the interaction of two (or more) devices is secured by an authenticated key exchange, where authentication usually means *entity* authentication. However, we feel that security, based on entity authentication, is likely to be inadequate in the pervasive computing paradigm, for two principle reasons:

- Names of entities will probably be unknown – a rather fundamental obstacle for entity authentication!
- Authenticating an entity (supposing that its identity can be reliably determined) is not likely to give us much confidence about what that device will *do*.

The motivating examples, which reveal the limitations of entity authentication, and indicate how entity authentication should be revised include: remote controls for home appliances [1], public information utilities (*e.g.* printers in airports, [2]), public wireless web access, teleworking and m-commerce. Literature discussing these new scenarios is gradually emerging. When using a public printer to print a secure document, via a wireless link, we are likely to be concerned that the data is received by only one device, *viz.* our chosen printer. Furthermore, we will probably want some assurance that our secure data will not subsist (especially as plaintext), after we have finished. We hope this example makes clear that device names, in pervasive computing, are likely to be of little relevance, as well as being indeterminate.

Reflecting on this example does yield some clues about how we might revise traditional notions of authentication. A name is an *attribute* of an object, and for pervasive computing security it may not be a very important attribute. But what about other attributes of objects? In the case of the printer it seems to us that *location* is an important attribute, physical location should be enough to specify a device like a printer. In addition to location we might be more confident about using an arbitrary printer were we sure that it had been manufactured by a reliable company, so this may also be an important attribute. For devices in general the *type of device* is also likely to be an important attribute.

The ‘state’ of a device is also one of its attributes and aspects of the state of a device are likely to be relevant to security. Even though a device has been authenticated to originate from a reliable source, we still need to know that nothing has been done to it subsequently, that might compromise its security. For instance, we may want to be sure that routine maintenance has not compromised its original reliability. Another important aspect of state might be that the device is not running another concurrent session, with someone else.

In general, by authenticating various attributes we would aim to confirm precisely *which* devices are the subjects of an interaction and *what* those devices are doing, or will do. Clearly, there may be a collection of attributes that we want to authenticate, and the context and level of assurance we want will determine the elements of the collection. However, in pervasive computing multiple factors will determine the levels of assurance required, these various factors will conceivably form a matrix. One dimension will be the threats that we want to guard against and the criticality of the security service that we want to protect. The threats will vary from juvenile hacking, to corporate espionage, right up to cyber terrorism and government surveillance. Another dimension will be the type of association that is being made, this will vary from a transient link (as is the case with a public printer), to on-going associations (with devices in the home or office), to long term or lifetime associations (for instance, a pacemaker). Just as group key management is significantly more complex than the two party case, multiparty associations will exacerbate the complexity of security requirements.

¹ SAG, QinetiQ Malvern. {I.Zakiuddin,S.Creese}@eris.QinetiQ.com

² Oxford University Computing Laboratory. Bill.Roscoe@comlab.ox.ac.uk

³ Formal (Europe) Systems Ltd and Worcester College, Oxford. michael@fsel.com

Thus far we have only discussed security requirements, we'd also like to present a few thoughts about how security may be enforced. It seems to us that just as the requirements will vary greatly, depending on the context, so the security mechanisms will, potentially, be very diverse.

Firstly, we should think about traditional methods of authentication, *viz.* certificates and signatures. Debate about the practicality of PKI's (for instance, the SHAMAN [3] project) is an on-going subject. This debate should be extended to the use of certificates, and the value they add, in the pervasive paradigm. What if we have to assume that Trusted Third Parties are not always accessible? In this case what trust will certificates carry? And how will they be used? Will certificates have to be issued per device, for its lifetime? Despite these questions, signatures may be useful to confirm static attributes like the origin of an object and its type. For the problem of trusted maintenance, assurance may be possible with a timestamped signature from a trusted maintainer.

One important use of certificates, in conventional security, is preventing man-in-the-middle attacks, but to do this the certificate binds to a name. In the light of our earlier discussion, about names being indeterminate, how will man-in-the-middle attacks be prevented? If it were possible to bind certificates to attributes, which uniquely specified a device, then this may provide an alternative use of existing prevention techniques.

Mechanisms for authenticating which devices are the subjects of an interaction are likely to operate at the level of the communications network. For instance, it may be possible to avoid man-in-the-middle attacks by constraining the resources that may be used in a transaction. A reliable GPS server may be able to locate devices, but then the assurance will depend on a secure GPS link; furthermore, for some applications, the location resolution may be insufficient (for instance, at the PAN level). Sound instrumentation of networking mechanisms, to authenticate which devices, will be non-trivial.

To give assurance about what devices will do, a variety of mechanisms might be deployed. And, in contrast to authenticating which, authenticating what seems to demand mechanisms at the device level. For instance, electronic devices might be engineered so that tampering destroys security information. The mobile code community has studied extensive techniques for *self-certifying* code. Security for pervasive computing could use similar concepts (it is not clear how much will simply 'port'), for instance hardware could be configured to send a hash of its configuration, then policies may be implemented where devices will only interact with other self-certifying devices. If the assurance at the hardware level was sufficiently high, then this could be used to mitigate the assurance demanded at the network level.

Continuing to draw inspiration from the mobile code community, user friendly security could be enforced by having a mobile agent act on the users behalf. A user's agent might certify that other devices are fit for interaction, in the sorts of sense that we have discussed. Such approaches would have the obvious drawback of needing to ensure that the agent was running correctly on the correct device. It may be observed that the 'which' and the 'what' have simply been shifted, but an agent approach may broaden the range of techniques that can be deployed, as well as yielding user friendly solutions.

Before concluding we must note that effective technology cannot be divorced from social and legal factors. So, whatever requirements and mechanisms are developed and deployed, they must either leave users with confidence that their security needs are safeguarded, or (perhaps equivalently) they must engender sufficient traceability to support litigation. In conclusion, we hope that this position paper has made clear both the breadth and the depth of the problem of security in pervasive computing. More specifically, we also hope it is clear that authentication, revised on the lines discussed, will be at the core of security in this new paradigm. Finally, we'd like to cram in some thanks to Peter Ryan and Gavin Lowe, for some very stimulating discussions; and to Chris Mitchell, for inviting us!

References

1. Stajano, F. and R. J. Anderson. The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks. In *7th Security Protocols Workshop*, LNCS vol. 1796, Cambridge, UK.
2. Balfanz, Dirk, D. K. Smetters, P. Stewart and H. Chi Wong. Trusting Strangers: Authentication in Ad-hoc Wireless Networks. Network and Distributed Systems Security Symposium, 2002. Available at: <http://www.isoc.org/isoc/conferences/ndss/02/proceedings/index.html>
3. <http://www.ist-shaman.org/>