SECURITY, FAULT-TOLERANCE AND THEIR VERIFICATION FOR AMBIENT SYSTEMS*

Jaap-Henk Hoepman
Department of Computer Science, University of Nijmegen
P.O. Box 9010, 6500 GL Nijmegen, the Netherlands
jhh@cs.kun.nl

Abstract For the emerging ambient environments, in which interconnected intelligent devices will surround us to increase the comfort of our lives, fault tolerance and security are of paramount importance. In contrast to the computers in a normal distributed system, ambient devices are generally small (meaning they have little computing power or memory space), often battery operated and interconnected much more dynamically. In this paper we discuss the fundamental research issues that emerge while designing the distributed algorithms for such ambient systems that must be both fault tolerant and secure.

Keywords: Security, cryptography, fault tolerance, formal methods, program verification, dependability, autonomous systems, distributed systems, ambient intelligence, embedded systems.

1. Introduction

In order to reduce cost, and to make management and addition of new services easier, our critical infrastructures (like the energy grid, or telecommunication systems) are increasingly built on and interconnected with Internet technology. Because of the open and best-effort nature of the Internet, these infrastructures have become increasingly vulnerable both to malicious attacks and ordinary faults [PCC97], implying that not only security, but also fault tolerance should be a real concern [Eld01].

Similar problems trouble the vision of future ambient environments [RTD02], in which interconnected intelligent devices will surround us, at home or while travelling. With these intelligent devices integrated in all kinds of consumer electronics (ranging from light switches, thermostats to audio, television and kitchen equipment), people expect equivalent reliability levels. Hence, security (to

*Id: sftv-position.tex,v 1.2 2003/01/14 09:08:56 hoepman Exp

protect the home environment from hostile interactions with the outside world) and fault tolerance (to gracefully handle unavoidable system and component failures) need to be incorporated into the ambient, autonomous, environment, as well as privacy enhancing technologies [Hor01].

The last few years more and more embedded systems have become remotely managed over Internet connections, without much consideration for security and fault tolerance issues [Ano02]. This is clearly of great concern. As explained by Wood *et al.* [WS02], even denial of service attacks in sensor networks may permit real-world damage to the health and safety of people, when these devices are deployed in health-care and safety critical applications.

Emerging research is addressing the combined treatment of both fault tolerance and security [MM99]. However, current research topics are scattered, and do not address the fundamental issues concerning a unified treatment of security and fault tolerance in protocol design, and the verification of these properties. Moreover, the proposed solutions do not scale very well to the ambient world, either because the complexity rises sharply with the number of nodes in the system (which in an ambient system may be in the order of billions), or because they do not consider the resource poorness inherent to the embedded devices.

In this paper we discuss the fundamental research issues that emerge while designing distributed algorithms for ambient systems that must be both fault tolerant and secure. For concreteness, we also briefly describe some of our specific research topics in this area.

2. Ambient systems: fundamental research issues

Ambient systems posses some unique characteristics that set them apart from current large scale distributed systems and networks, posing new fundamental research questions in the area of fault tolerance and security.

2.1 Characteristics of ambient systems

An ambient system consists of many small interconnected devices. On a global scale, the number of such devices may easily run into the billions. Devices may range from special purpose controllers to general purpose computers, with special purpose but programmable devices (like mobile phones) or other general purpose handheld devices somewhere in between. In general, these devices should be cheap and therefore kept as simple as possible. They contain just enough computing power and storage space to accomplish their tasks. Some may not even have a form of persistent storage. Many of them will not be powered directly, but instead are powered through e.g., batteries or (either because they are portable, or because they are in a location without access to the power grid). To summarise, an ambient device is *resource poor*: it has little memory, only a small CPU, and should consume little energy.

The devices in an ambient system are interconnected in many different ways. Many only have a simple short range wireless interface (e.g. Bluetooth or infrared). Others may be equipped with larger range wireless interfaces (e.g. GSM/UMTS, or WiFi) or a fixed network connection. With such diverse network connections, and the highly mobile nature of at least some of the devices, we see that the ambient network is highly dynamic in its topology and exhibits unpredictable availability. Moreover, there is no hope to achieve any kind of centralised control.

2.2 Fundamental research questions

These unique characteristics of ambient systems give rise to the following largely open new fundamental research questions.

The traditional measures to analyse distributed algorithms have been *message complexity* (either in the number of messages sent or in the number of bits transmitted) and *time complexity*. For mostly battery powered ambient systems, energy consumption is also great concern. Therefore, the *energy complexity* of an algorithm should be analysed as well. Clearly, energy complexity of an algorithm is strongly related to both its time and message complexity. But this relationship is not as straightforward as it may seem at first.

- Sending messages consumes more energy than receiving. For broadcast channels like wireless networks, broadcasting a message takes as much energy as sending it to any particular node within transmission range.
- Waiting for a message ("listening to the wire") consumes considerable energy, yet waiting for a message is usually not included in the total time complexity of an algorithm.
- In emerging energy conserving reconfigurable hardware platforms hardware reconfiguration may either speed up the computation or conserve energy.
- The size of the data structure used by the algorithm influences its energy consumption: they may require secondary storage (either discs, EEPROM, or auxiliary RAM) that could otherwise have been switched off.

A realistic yet generally applicable and future proof model of the energy consumption of a computing device – the "battery powered" Turing Machine – is required [LV92]. Within such a model, we can reason about and maximise *security and fault-tolerance per Joule*.

In cryptography, the strength of a cipher is most realistically measured in terms of its *security parameter* k [Gol97][Bel98]. Within this model, ciphers and protocols are shown to require at least $g(k)$ work to break by the adversary (where a protocol is considered secure iff g is super polynomial in k). In

other words, the probability of the attacker to break the system in one particular instance is $o(1/g(k))$, i.e., negligible in k for secure protocols. This is a weaker condition than the usual requirement for fault tolerant algorithms that must remain operational whatever the actions and work spent by the adversary (a condition that can usually only be met by restricting the number of faulty nodes f to be significantly less than the total number of nodes n) [LSP82]. For probabilistic fault tolerant algorithms, the usual approach is to maintain impossibility of failure, and to use randomisation to improve the expected running time [Rab83].

When studying algorithms that need to offer both security and fault tolerance, it does not make sense to keep this strong notion of fault tolerance. Instead the level of fault tolerance should be comparable to the strength of the security measures, and should therefore be expressed in terms of the security parameter k as well. Moreover, the traditional Byzantine failure model assumes that the adversary has unlimited computing resources and can guess or modify any messages sent by the nodes. This is too strong because it would defy any cryptographic security protection. A model similar to the authenticated Byzantine agreement systems (where the adversary is unable to forge signatures) [DS83] appears to be more suitable. In any case, this approach to specifying the security and fault tolerance of protocols poses new challenges to their formal verification, where cryptographic primitives are usually considered as black boxes that offer absolute security.

Finally, the lack of any centralised control and the fact that the network is highly dynamic (both in overall structure and connectedness of individual nodes) has large ramifications for the design of secure protocols. For instance, many security protocols require the assistance of a trusted third party. In practice even in fixed networks with centralised control it turns out to be hard to find parties that most agree on to be trustworthy. For highly dynamic networks, this is even harder. Moreover, in such networks trusted third parties may often be unreachable. Also, Public Key Infrastructures (PKI) with their hierarchical structure of certification authorities appear to be inappropriate for ambient networks. This is not only caused by their dynamic network topology, but also because the concept of ownership of and control over the embedded ambient devices becomes much more diverse and diffuse.

3. Emerging research topics

We conclude this paper with the description of three concrete examples of emerging research topics that we plan to study in this area in the coming years.

A challenging topic to focus our research on is the design of a fault tolerant and secure Domain Name System (DNS) [AL01]. Because of the large dependence of many Internet applications on the DNS, proper functioning of the

DNS is of vital importance, and in fact proposals for and first implementations of a secure version of DNS (called DNSSEC) are emerging [Eas99]. However, current DNSSEC proposals suffer from two shortcomings. Due to backward compatibility requirements, DNSSEC has to be compatible and interoperable with older, non-secure, DNS clients and servers. This has seriously limited the available design options, adversely influencing the security of the overall design. Apart from the existence in the traditional DNS of 13 root domain servers, and the use of caching to speed up serving requests DNSSEC does not address the issue of fault tolerance at all. By dropping the backward compatibility requirement, we aim to study the ideal design of a secure and fault tolerance DNS. An extra challenge is to make sure the design can also be used within resource poor ambient environments that will (just as much as the current Internet) rely on a stable domain name system [Gie01].

We are currently raising support to investigate fault tolerant privacy protection of users in an ambient world [EHL+02]. Here we aim at a two-tier approach. In order to prevent collection of data about the user a system that allows for pseudo-identities and secure computing mechanisms will be investigated. On the other hand, to control the dissemination of data of the user, a license-based system will be developed. The licensing system will be based on the premise that user data needs to be accompanied by a license that shows that a party obtained the data rightfully. A license prescribes the actions (such as viewing, linking or transferring) a party is allowed to perform on the data. This requires a careful drafting of the semantics (i.e. meaning) of such licenses, efficient structuring and encoding of licenses, as well as protocols to securely issue, verify, transfer, split, limit, and combine licenses. Throughout the project, solutions must be both fault tolerant and secure, and are constrained by the limited computing, storage and communication resources available on the ambient devices, while making use of the much larger resources available in the home gateways and beyond.

Another interesting avenue of research is the issue of key management and key distribution in faulty environments [SMF+02]. The goal here is to devise secure key distribution protocols (for example those used in group membership protocols), that can not only survive processor failures but also memory/state corruptions. The latter type of errors have been extensively studied in the area of *self-stabilisation* [Dij74]. Algorithms that are both self-stabilising and fault tolerant are much harder to design [AH93][HPT02], and this is even without any security requirements.

References

[AL01] ALBITZ, P., AND LIU, C. *DNS and BIND*. O'Reilly & Assoc., Inc., 2001.

[AH93] ANAGNOSTOU, E., AND HADZILACOS, V. Tolerating transient and permanent failures. In *7th WDAG* (Lausanne, Switzerland, 1993), A. Schiper (Ed.), LNCS

725, Springer-Verlag, pp. 174–188.

[Ano02] ANONYMOUS. Embedded systems. Crypto-gram, December 15, 2002.

[Bel98] BELLARE, M. Practice-oriented provable-security. In *Proceedings of First International Workshop on Information Security (ISW 97)* (1998), E. Okamoto, G. Davida, and M. Mambo (Eds.), LNCS 1396, Springer, Berlin.

[Dij74] DIJKSTRA, E. W. Self-stabilizing systems in spite of distributed control. *Comm. ACM* **17**, 11 (1974), 643–644.

[DS83] DOLEV, D., AND STRONG, H. R. Authenticated algorithms for byzantine agreement. *SIAM J. Comput.* **12**, 4 (1983), 656–666.

[Eas99] EASTLAKE, D. Domain name system security extensions. Tech. Rep. RFC 2535, IETF, 1999.

[Eld01] ELDER, M. *Fault Tolerance in Critical Information Systems*. PhD thesis, Department of Computer Science, University of Virginia, 2001.

[EHL+02] ETALLE, S., HOEPMAN, J.-H., LUBBER, J. C. A. VAN DER, VERSCHUREN, J. H. S., AND HUIZENGA, J. PAW: Privacy in an Ambient World. GENCOM project proposal, 2002.

[Gie01] GIEBEN, R. Chain of trust: The parent-child and keyholder-keysigner relations and their communication in dnssec. Tech. Rep. CSI-R0111, University of Nijmegen, The Netherlands, 2001.

[Gol97] GOLDREICH, O. On the foundations of modern cryptography. In *CRYPTO '97* (Santa Barbara, CA, USA, 1997), B. S. K. Jr. (Ed.), LNCS 1294, Springer.

[HPT02] HOEPMAN, J.-H., PAPATRIANTAFILOU, M., AND TSIGAS, P. Self-stabilization of wait-free shared memory objects. *J. Parallel & Distr. Comput.* **62**, 5 (2002), 766–791.

[Hor01] HORN, P. *Autonomic Computing*. IBM Corp., 2001.

[LSP82] LAMPORT, L., SHOSTAK, R., AND PEASE, M. The byzantine generals problem. *ACM Trans. Prog. Lang. & Syst.* **4**, 3 (1982), 382–401.

[LV92] LI, M., AND VITÁNYI, P. M. B. Theory of thermodynamics of computation. In *PROC. IEEE Phys. of Comput. Workshop* (Dallas, TX, USA, 1992), pp. 42–46.

[MM99] MEADOWS, C., AND MCLEAN, J. Security and dependability: then and now. In *Computer Security, Dependability, and Assurance: From Needs to Solutions, 7-9 July 1998 & 11-13 November 1998, York, UK & Williamsburg, VA, USA* (1999), Los Alamitos, CA, USA : IEEE Comput. Soc, 1999, pp. 166–70.

[PCC97] Critical foundations: Protecting America's infrastructure. Report of the President's Commission on Critical Infrastructure Protection, 1997.

[Rab83] RABIN, M. O. Randomized byzantine generals. In *24rd FOCS* (Tucson, AZ, USA, 1983), IEEE Comp. Soc. Press, pp. 403–409.

[RTD02] The priorities of the sixth framework programme 2002–2006. RTD Info, DG Research, European Commission, 2002.

[SMF+02] STADDON, J., MINER, S., FRANKLIN, M., BALFANZ, D., MALKIN, M., AND DEAN, D. Self-healing key distribution with revocation. In *IEEE Security & Privacy* (Berkeley, CA, USA, 2002), IEEE, pp. 241–257.

[WS02] WOOD, A. D., AND STANKOVIC, J. A. Denial of service in sensor networks. *IEEE Comput.* (2002), 54–62.