Electronic Notes in Theoretical Computer Science 20 (1999)
URL: http://www.elsevier.nl/locate/entcs/volume20.html 4 pages

The MFPS XV Security Session

Catherine Meadows

Naval Research Laboratory
Code 5543
Washington, DC 20372
meadows@itd.nrl.navy.mil

Dennis Volpano

Department of Computer Science
Naval Postgraduate School
Monterey, CA
volpano@cs.nps.navy.mil

1 Introduction

Security has long been a popular application of formal methods. This is because it is a fertile source of challenging problems that are important enough to justify the effort involved in developing mathematical models and formal techniques. And their importance is growing. We are moving to a more networked world where our vital transactions depend upon our ability to communicate securely over an untrusted network and upon information and software obtained from parties about whom we may know little if anything. To meet these challenges, MFPS is bringing people in formal methods and semantics together with researchers in the field of security. A special session of MFPS15 was devoted to security. It involved one invited talk by Martín Abadi, and six speakers, Dominique Bolignano, Carl Gunter, Pat Lincoln, George Necula, Geoffrey Smith, and Paul Syverson. The speakers covered four major areas of security. In this introduction, we give an overview of these areas and indicate why they are important and what makes them difficult. We also give a brief outline of the speakers' talks.

2 Cryptographic Protocol Verification

In order to communicate securely over an insecure network, it is necessary to use encryption to provide secrecy and to authenticate messages, and to develop protocols that use cryptography to perform such functions as the distribution of keys and the authentication of principals and transactions. But the use of cryptography does not in itself guarantee correctness; in many cases it may

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
01 SEP 1999	**N/A**	**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
The MFPS XV Security Session	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Naval Research Laboratory Code 5543 Washington, DC 20372**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**UU**	**4**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

be possible for a hostile intruder who has the ability to read, redirect, and alter messages to manipulate the protocol into revealing secret information or allowing the intruder to impersonate an honest principal, without breaking the underlying crypto-algorithms. This concern is not merely a theoretical one; numerous examples exist of protocols that were at one time believed to be secure but were found out to have serious security flaws some time after they were published[1]. In their talks, Bolignano, Syverson and Lincoln each addressed different aspects of this problem.

Bolignano described work related to his analysis of electronic commerce protocols. Such protocols are typically very complicated, and the security properties proved typically involve proving the integrity of complex data structures. Thus it is necessary to find *safe abstractions*, that is abstractions that reduce the complexity of the system to be analyzed without jeopardizing the correctness of the conclusions reached. Bolignano outlined his techniques for finding such safe abstractions for cryptographic protocols.

Syverson addressed the problem of reconciling belief logics developed for the analysis of cryptographic protocols, which require a relatively small amount of computational effort but tend to be overly abstract, with state-based models, which are more detailed (and hence usually more accurate), but whose use in analysis tends to be more computationally intensive. He used the recently introduced strand space model [2], that ties together much of the recent work in state-based cryptographic protocol analysis, to provide a semantics for the modal authentication logic SVO [6].

Lincoln described a framework for analyzing security protocols in which protocol adversaries may be arbitrary probabilistic polynomial-time processes. In this framework, protocols are written in a restricted form of a the π-calculus [5], a formal specification language developed for reasoning about communication in distributed systems, and secrecy is formulated in terms of observational equivalence which involves quantifying over the possible environments that can interact with a protocol. This allows a more accurate model of the role cryptography plays in a cryptographic protocol while still retaining the benefits provided by a formal specification language. He also mentioned some more recent results in the complexity of analyzing secrecy in simple cryptographic protocols. The problem of determining whether a protocol allows an intruder to gain access to a given secret is undecidable even for protocols with very strong restrictions on various parameters like message length and nesting depth of encryption.

3 Public Key Infrastructure

Public key cryptography provides a powerful authentication mechanism. A principal can sign a message with a private key that only it knows, and any-

[1] See [1] for a few examples.

one can verify it with the corresponding public key. But this alone is not very useful unless there is a way of associating public keys with principals. The earliest work on public key cryptography suggested that public keys be published in a central place, such as a telephone directory. However, with the growing widespread use of public keys, this is no longer practical. The common use now is to have a public key authority that signs (and thus vouches for) a certificate containing the principal's name and the public key belonging to it. The public key of this authority may be signed by a higher authority, and so forth, so that a public key hierarchy is obtained. The issue is complicated by the fact that many different hierarchies may exist, that circular chains of authentication may be allowed (e.g. the PGP "web of trust"), that certificates may be used not only to provide authentication of keys but to specify different privileges belonging to principals, and that both keys and privileges may be revoked by an authority. It is necessary to develop a sound and expressive logic to reason about policies in this framework and describe them without ambiguity. In his talk, Gunter showed how type theory can be applied to the problem of certificate revocation and addressed the issues raised by the non-monoticity that such revocation introduces.

4 Secrecy Models

The ability not to reveal sensitive information is a key feature of security. However, it usually is not practical to verify that every piece of code that has access to secret information is trusted not to reveal it. Instead, it is quite common to have some smaller part of a system enforce a security policy describing the types of communication a process with access to sensitive information may have with other parts of the system. However, when the stakes are high, a simple access control policy may not be enough. A Trojan Horse in the untrusted process could use any visible effect the process has on the system as a covert channel in which the sensitive information could be encoded. Visible effects could include resources used by the process, delays in processing for other parts of the system (timing channels) and even changes in the probability that other events would or would not occur. The problem was first noted by Lampson in 1973 [4], and has motivated much of the research in multilevel security, which deals with the problem of maintaining separation between data classified at different security levels in the same system. This problem has remained with us even as we move from timesharing operating systems to more networked architectures [3]. In his talk, Smith described a model for secrecy that takes into account, not only a process's ability to produce events that may be seen by another process, but a process's ability to affect the *probability* of certain events.

5 Code Verification

Correctness of code has always been an important problem. But code verification, although it got off to a promising start, has in recent years been regarded as too difficult to be practical, and efforts instead have concentrated on verification of higher-level system specifications. But increasing use of mobile code, that is, code which is sent over the network and executed, has sparked new interest in developing the best possible methods of assuring the safety of the code itself, by the user as well as the developer, and doing so in an automated way. Necula described the concept of proof-carrying code, in which mobile code carries its own proofs of safety with it, which can be mechanically verified by the target execution environment.

6 Conclusion

Tools and techniques developed as part of the foundations of programming languages and their logics can be applied fruitfully to some aspects of security. The speakers in the Security Session provided yet more evidence of the growing synergy between the semantics of programming languages and security. For the details of their work, we invite you to read the papers that appear in these proceedings.

References

[1] M. Burrows, M. Abadi, and R. Needham. A logic of authentication. *ACM Transactions on Computer Systems*, 8(1):18–36, February 1990.

[2] F. Javier Thayer Fabréga, Jonathan C. Herzog, and Joshua D. Guttman. Strand spaces: Why is a security protocol correct? In *Proceedings of the 1998 IEEE Symposium on Security and Privacy*, pages 160–171. IEEE Computer Society Press, May 1998.

[3] Myong H. Kang and Ira S. Moskowitz. A network pump. *IEEE Transactions on Software Engineering*, 22(5), May 1996.

[4] B. W. Lampson. A note on the confinement problem. *Communications of the ACM*, 16(10):613–615, October 1973.

[5] R. Milner, J. Parrow, and D. Walker. A calculus of mobile processes, parts I and II. *Information and Computation*, 100:1–77, September 1992.

[6] Paul F. Syverson and Paul C. van Oorschot. On unifying some cryptographic protocol logics. In *Proceedings of the 1994 Symposium on Security and Privacy*. IEEE Computer Society Press, May 1994.