Artificial intelligence and security: An overview

by ALAN C. SCHULTZ
The Navy Center for Applied Research in Artificial Intelligence
Washington, District of Columbia

ABSTRACT

The junction of AI and computer security is an area of increasing concern, due to the imminent application of AI to fielded systems. Two new areas of research need are identified: artificial intelligence techniques in the development of secure systems and in analyzing the security characteristics of software; and verification of the security of artificial intelligence. Current and proposed research in these areas by the Department of Defense will be discussed.

INTRODUCTION

While the areas of artificial intelligence and computer security have been explored for many years, the intersection contains many interesting, useful, and, in some cases, dangerous implications. The intersection can be viewed from two directions. First, how can artificial intelligence techniques be used in the design and analysis of secure systems? Second, what can be said about the security characteristics of artificial intelligence software, particularly expert systems?

Artificial intelligence techniques are being relied on more in various security related tasks. Although some work has been done in both directions, the intersection still has many under-explored or unexplored areas in need of further research. This paper will briefly identify some areas under research, and areas in need of exploration.

USING AI TECHNIQUES IN COMPUTER SECURITY DESIGN AND ANALYSIS

While many software engineering tools and methodologies have been devised to help in creating reliable and easy to maintain software, secure software and systems require a greater degree of assurance about their behavior. One area that has received much attention is in formal verification of software. The necessity of formal verification is mandated by the National Computer Security Center (NCSC), which requires that for a computer system to achieve a top rating of A1, a formal top-level proof must be done for the system.[1] Artificial intelligence techniques have been introduced in the form of automated theorem provers.

Given a program and a set of formal specifications, an automatic theorem prover can be used to verify that the program satisfies the specifications. One example of a verification system that uses a theorem prover is Gypsy.[2] Although other verification systems are in use, Gypsy has been used with much success, particularly by NCSC.

Looking towards the future, it has been said that the ultimate goal of artificial intelligence applied to software engineering is automatic programming, and we might expect to have a system that automatically generates secure software when a user specifies the requirements.

While the above methods are useful in the development stage of software, experience has shown that they cannot be applied to existing software. Large bodies of software exist that need to be used in secure environments. Therefore, testing and analysis techniques are used to determine the security characteristics of the software. In this area, very little work has been done using artificial intelligence techniques.

As far back as 1974, the RISOS (Research in Secure Operating Systems) project at Lawrence Livermore Laboratory had developed a set of tools to analyze operating systems for security flaws.[3] The tools used powerful pattern-matching techniques to search the code for sequences of operations that might characterize security flaws. The tools analyzed various assembly languages, and are not currently in use; although at the time good results were obtained. The tools should be updated to analyze high-level languages.

One area that could be quite productive is the use of an expert system to analyze software and recommend testing strategies—a task well suited for an expert system. In this respect, the expert system would act as an assistant to a security analyst.

A related issue is the study of a system in operation to discover security violations. In this area, several groups have made advances using artificial intelligence techniques for intrusion detection and for on-line analysis of the system.

Discovery is the name of TRW's expert system that is used to detect anomalies in subscribers' usage of a database. The system searches for frequently occurring patterns in data and compares these patterns to daily activity to detect variations from normal behavior.[4] Sytek, under contract for the Department of the Navy, is investigating the use of pattern matching for the automated analysis of audit trails to assist security officers in detecting security violations.[5] Still others are using pattern matching and audit trails for intrusion detection.[6,7]

SECURITY OF EXPERT SYSTEMS

The other side of the artificial intelligence and computer security coin is an area of much concern. Specifically, what can be said about the security characteristics of artificial intelligence programs, in particular, expert systems. Now that expert systems are starting to be routinely created and used, computer security officers must now concern themselves with the security analysis of these systems.

Although in the early days of expert systems, they were hailed as being easy to maintain and understand, most would now agree that expert systems are actually hard to understand and maintain. The existing methodologies for software design and maintenance are not readily applied to expert systems, and this is one area that needs considerable research.

At least one research group is currently investigating design methodologies for rule-based systems.[8] More work needs to be done in the verification of expert systems in order to assure their behavior prior to installation in a security environment.

Other areas of artificial intelligence research will have even greater difficulties with computer security. What of systems that learn? There must be some assurance that these systems maintain their security characteristics. No research to date has addressed this problem, since machine learning is still in its infancy. However, the problem should be addressed

now, and should not wait until systems have been implemented and installed.

CONCLUSION

Artificial intelligence techniques are starting to be applied to the analysis of secure computer systems, and, hopefully, their use will improve the utility of security analysis and verification. On the other side of the coin, more research is needed to address the security implications of artificial intelligence systems. Design, verification, and analysis techniques are needed for expert systems and systems with learning mechanisms, and these techniques should be developed now, not after the systems are fielded.

ACKNOWLEDGEMENTS

I would like to thank Randall Shumaker and Carl Landwehr for their time and insight and David Rine for the opportunity to express myself.

REFERENCES

1. "Trusted Computer System Evaluation Criteria." CSC-STD-001-83, Department of Defense, August, 1983.
2. Good, D.I. "Mechanical Proofs About Computer Programs." *Philos. Transactions of the Royal Society of London,* 312 (1984) 1522, pp. 389–409.
3. "Handbook for Analyzing the Security of Operating Systems." RISOS Project. DOD S5-2068, November, 1976.
4. Tener, William. "Discovery: An Expert System in the Commercial Data Security Environment." *Information Security: The Challenge,* IFIP, Reprints of the Fourth IFIP Security on Information Systems, 1986, pp. 283–291.
5. Halme, L. and J. Van Horne. "Automated Analysis of Computer System Audit Trails for Security Purposes." *Ninth Annual National Computer Security Conference Proceedings,* NCSC, Washington, D.C., September, 1986, pp. 71–74.
6. Denning, Dorothy and Peter G. Neumann, "Requirements and Model for IDES—A Real-Time Intrusion-Detection Expert System." Menlo Park, California: SRI International, August, 1985.
7. Kuhn, J. "Research Toward Intrusion Detection Through Automated Abstraction of Audit Data." *Ninth Annual National Computer Security Conference Proceedings,* NCSC, Washington, D.C.: September, 1986.
8. Jacob, Robert J.K. and Judith N. Froscher, "Software Engineering for Rule-Based Systems." *Proceedings of the Fall Joint Computer Conference,* IEEE Computer Society Press, November 1986, pp. 185–189.