



MONTCLAIR STATE
UNIVERSITY

Montclair State University
**Montclair State University Digital
Commons**

Department of Computer Science Faculty
Scholarship and Creative Works

Department of Computer Science

12-1-2005

Distributed Computing and Computer Security Education

Stefan Robila

Montclair State University, robilas@mail.montclair.edu

Follow this and additional works at: <https://digitalcommons.montclair.edu/compusci-facpubs>



Part of the [Computer Sciences Commons](#)

MSU Digital Commons Citation

Robila, Stefan, "Distributed Computing and Computer Security Education" (2005). *Department of Computer Science Faculty Scholarship and Creative Works*. 231.

<https://digitalcommons.montclair.edu/compusci-facpubs/231>

This Conference Proceeding is brought to you for free and open access by the Department of Computer Science at Montclair State University Digital Commons. It has been accepted for inclusion in Department of Computer Science Faculty Scholarship and Creative Works by an authorized administrator of Montclair State University Digital Commons. For more information, please contact digitalcommons@montclair.edu.

Distributed Computing and Computer Security Education

Stefan A. Robila
Montclair State University
RI 301
Montclair, NJ, 07043
(973) 655-4230

robilas@mail.montclair.edu

ABSTRACT

This paper presents our experience in using cluster computing when teaching Computer and Data Security. The background knowledge together with the topics that fit the best applications of distributed computing and the hardware and software needs are discussed. Several project activities are presented with some analyzed in detail. The first requires the students to develop a computer cluster out of a regular public lab and use it for building security attacks such as password and encryption key cracking. The second deals with prime number generation using a client /server architecture (implemented in Java) and an Oracle database complemented by the implementation of resiliency (reassigning jobs when a node dies). These projects allow the implementation of real world – like attacks while using relatively inexpensive resources. Based on class assessment, they were highly appreciated by the students, increasing their awareness on the power of distributed computing with respect to data security and ensuring their preparedness to further advances in computing.

Categories and Subject Descriptors

K.3.3 [Computer and Education]: Computer & Information Science Education – *Computer Science Education, Curriculum*. C.2.0 [Computer-Communication Networks]: General - *Security & Protection*, D.4.6 [Operating Systems]: Security and Protection - *Authentication*. K.6.5 [Management of Computing & Information Systems]: Security & Protection Education

General Terms

Algorithms, Design, Performance, Security

Keywords

class projects, prime numbers, passwords, inexpensive computing

1. INTRODUCTION

Traditionally, computer security applications are relying on the time complexity of mounting attacks against them. Barring any weaknesses in the techniques themselves, the strength of an algorithm is assessed against the number of efforts to break it, usually comparable to brute force. Given the large size of the data (i.e. key or search space), brute force was initially thought as usable only by supercomputers that required significant cost and time investment. In the last decade, an increased development of cluster

computing has shown that extensive attacks can be mounted using networks of inexpensive computers prompting a need to reassess the time complexity of security attacks [1]. While scientific research abounds in this area, references and activities linking distributed computing to security are still undeveloped in IT education [2].

In the following we present examples of computer security projects that can be implemented by students enrolled in topics courses or through independent study. Some of these projects were developed in a computer security course taught within our CS program and resulted in student presentations at local conferences. In addition, the individual experiences, wereshared with peers through class presentations. The paper is organized as follows. In section 2 we briefly discuss the nature and content of the course. In section 3 we present the project suggestions and discuss other implications such as ethics and resources that are needed. The paper ends with conclusions (section 4) and references.

2. COMPUTER AND DATA SECURITY

The course is a survey of topics related to computer security. It introduces the students to many contemporary topics ranging from data encryption such as PKIs (Public Key Infrastructures), computer authentication, network security, to cyber-warfare and security ethics. Students will learn fundamental concepts of security that can be applied to many traditional aspects of computer programming and computer systems design as well as in dealing in day to day activities such as accessing confidential information, protecting computer systems, etc.

As instructional objectives, the students are expected to critically analyze various security issues (in terms of their complexity, vulnerability to attacks, etc.) including: cryptography techniques (encryption types, secret and public key methods, key exchanges, etc.), network protection (topologies, types of threats, protection, etc.), OS security (memory protection, file protection, users, trusted OS), database security (reliability, sensitive data, etc.), program security (viruses, buffer overflows, trapdoors), enterprise security (policies, planning, etc.), ethics in protection, pseudosecurity (steganography, watermarking, etc.). The course was developed based on Pfleeger & Pfleeger's textbook [3] with some materials from [4], closely following NSA recommendations on terminology and content. An important component of the course constitutes the practical assignment and the final projects. Each include significant work both in application development as well as in writing. The course was offered as elective for upper level CS majors and graduate students, and constitutes the major security component of the program although other components are integrated throughout our curriculum similar to [5].

3. DISTRIBUTED COMPUTING

3.1 Building and using a cluster

Distributed computing constitutes an excellent approach in the implementation of security attacks. Within several days, and using only freely available software, a team of students were able to turn a regular computer lab in a powerful cluster. Twenty five IBM M50 (2.8 GHz Pentium 4, 512 MB of RAM) computer systems were connected over their existing Cat-5e Ethernet network using CHAOS v.0.7 OpenMosix boot disks patched to enable clustering. The students were next able to use the forkjohn utility to control the number of nodes that were running and applied a dictionary based password cracking application to test the efficiency of their system. While the design of the network is simplistic compared to current standards [6], the use of free software and resources without their modification constitutes a good example of the possible security attacks that IT professionals must take into account. Of particular interest is that the students used the computer lab outside the normal operation hours. No configuration modification was noticed during the normal operation hours.

3.2 Distributed Prime Number Generation

In a second project, a team has implemented a prime number generator using a Java based client/server architecture and an Oracle database. The project was derived from an initial assignment that asked the students to create prime number generators in the programming environment of their choice. Following, a discussion on the suitability of various environments was done. The advantage of using distributed computing relies on a significant speedup in this computation as well as in the rather inexpensive systems that can be used. The project included a prime calculating code, calculating the prime numbers within an allotted range, which is distributed among different client processors, and a master program running on our server that regulates the input to the clients, and stores the results obtained from the clients into the database. The application was complemented by the implementation of resiliency (reassigning jobs when a node dies). While prime number generation is a classic problem in computer security [7], the use of distributed computing exposes the students to new thinking patterns in assessing complexity of various encryption techniques.

3.3 Other Project Suggestions

Several other projects that use distributed computing can be envisioned. One such example is the use of distributed computing for computer forensics. A computer system's directory structure is mapped and distributed computing is used to search for files of interest such as images or documents. A second stage is the processing of the files of interest. A structure similar to the projects above can be successfully used in the implementation. Another possible project is the design of repeated network attacks from various cluster nodes, allowing for the use of multiple nodes and addresses. Identification of such attacks constitutes a project on its own [8]. In addition, the use of cluster computing can be given as possible solution in the design of reliable computing environments by allowing for resilience.

3.4 Educational and Ethics Considerations

The above projects were implemented by students as part of their work within the Computer and Data security Course described in Section 2. Each term project was presented in class and benefited from faculty and peer feedback. In addition, each project team

submitted an extended report detailing their work, resources and references used. The cluster and prime number generation project were also presented as part of the local Sigma Xi student conference.

An important component in teaching computer security is the ethics aspect [9]. In the context of distributed computing, where significant computing power is readily available, fair use of technology needs to be carefully addressed. The activities discussed above were designed with this in mind. For example, following the project presentation, techniques to counter password cracking were discussed in connection with the project described in section 3.1. In addition, the IT administrator's use of password crackers for detection of weak passwords within a network was emphasized as a positive tool.

4. CONCLUSIONS

We presented considerations on including distributed computing topics within the computer security education. While the need for computer and data security is evident through the CS and IT curriculum, the risk introduced by distributed computing is seldom analyzed in this context. Our approach is the introduction of distributed computing techniques through term projects. While the development is usually limited to few students, the extended public presentations and peer interaction allowed for the development of a common class experience. The use of existing systems in building clusters allows for inexpensive implementation. Student feedback indicated a renewed awareness on the risks and possibilities introduced by the use of distributed environments.

5. REFERENCES

- [1] Yang, T. A., Yue, K-B., Liaw, M., Collins, G., Venkatraman, J., T., Achar, S., Sadasivam, K., and Chen P., Design of a distributed computer security lab. *Journal of Computing in Colleges*, vol 20, no 1, 2004, 332-346.
- [2] Georgiev, I. K., and Georgiev I.I., A security model for distributed computing. *Journal of Computing in Colleges*, vol 17, no 1, 2001, 178-186.
- [3] Pfleeger, C.,S. Pfleeger, *Security in Computing*, Prentice Hall, 2003
- [4] Stallings, W., *Network Security Essentials*, 2nd Ed., Prentice Hall, 2003
- [5] Null, L., Integrating security across the computer science curriculum, *Journal of Computing in Colleges*, vol 19, no 5, 2001, 170-178.
- [6] Treese, W., How to build a supercomputer, *netWorker*, vol 8, no 4, 2004, pp 15-18.
- [7] Shade, E., Ready for prime time?, *Journal of Computing in Colleges*, vol 17, no 3, 2002, 282-289.
- [8] Scambray, J., McClure, S., and Kurtz, G. *Hacking Exposed*, 4th Ed, McGraw-Hill Professional Publishing, 2003
- [9] Botting, R., Teaching and learning ethics in computer science: walking the walk, *Proceedings SIGCSE*, 2005, pp. 342-346.