Secure Coding Practices & Automated Assessment Tools

Prof. Barton P. Miller, University of Wisconsin, bart@cs.wisc.edu
Prof. Elisa Heymann, University of Wisconsin, elisa@cs.wisc.edu

Security is crucial to the software that we develop and use. With the incredible growth of both Web, Cloud, and Grid services, security is becoming even more critical.

> **Securing your network is not enough! Every service that you deploy is a window into your data center from the outside world, and a window that could be exploited by an attacker.**

This tutorial is relevant to anyone wanting to learn about minimizing security flaws in the software they develop or manage. We share our experiences gained from performing vulnerability assessments of critical middleware. You will learn skills critical for software developers and analysts concerned with security.

Software assurance tools – tools that scan the source or binary code of a program to find weaknesses – are the *first line of defense* in assessing the security of a software project. These tools can catch flaws in a program that affect both the correctness and safety of the code. This tutorial is also relevant to anyone wanting to learn how to use these automated assessment tools to minimize security flaws in the software they develop or manage.

Tutorial length: Half day.

General description: Our tutorial focuses on the programming practices that can lead to security vulnerabilities, and on automated tools for finding security weaknesses. This tutorial features several interactive secure coding quizzes where the audience will be challenged to find as many vulnerabilities as they can in short code fragments. The quizzes are a synthetized version of vulnerabilities that we found in real Grid/Cloud software. What the audience finds (and does not find) will then be discussed.

The first major technical area of our tutorial is a presentation of the most common vulnerabilities found in middleware and services. Descriptions of each type of vulnerability are presented with examples. The examples show how each type of vulnerability occurs within code, pointing out how common usage patterns for system library routines, kernel calls, and common programming techniques can result in the vulnerability. The coding examples are presented in C, C++, Java, Python and Perl.

Along with the description of the vulnerabilities, we show how the vulnerability can be mitigated or eliminated through the use of specific programming and design techniques. An important part of our discussion of each vulnerable technique is a description of the thought processes used by the attacker in developing an exploit.

The second technical area of our tutorial is a presentation about automated assessment tools. We introduce the different types of analysis tools, how these tools work, their output and their limitations.

The next section of the tutorial explains how to use different commercial and open source tools for C/C++ and Java, using the SWAMP, and how to process the tools' output. We use simple test applications extracted from the NIST/NSA Juliet test suite, where each of these applications contain specific weaknesses, and the version of the same code with the weakness fixed. We show how users can benefit from the Software Assurance Marketplace-SWAMP (https://continuousassurance.org/), which is an open facility that allows users to scan their software with different tools without the burden of dealing with tool acquisition, installation, and configuration. Throughout the SWAMP users can access both commercial and open source software assessment tools.

Targeted audience: This tutorial is targeted at Cloud/Grid software developers wishing to minimize the security flaws in the software that they develop. It covers the defensive side of security – how to prevent problems by showing many types of vulnerabilities that occur in real code and what techniques can be used to prevent them, and how to use automated analysis tools to detect flaws in their software. *The target audience for this tutorial is anyone involved with the development, deployment, assessment, or management of critical software.*

Tutorial Goals and benefits: The goals for this tutorial are to teach developers to think about security issues, provide specific techniques for writing secure code, and provide them with the tools that they need to help improve the security of their code. The benefits are improved security and reliability of our common HPC infrastructure.

Content level: 50% beginner, 25% intermediate, 25% advanced.

Audience prerequisites: To gain maximum benefit from this tutorial, attendees should be familiar with the process of developing software and at least one of the C, C++ Java or scripting programming languages. This tutorial does not assume any prior knowledge of security assessment or vulnerabilities.

Outline of the Tutorial

1. Thinking like an attacker
2. For each of the following categories we will
   - 2.1. Description of vulnerability
   - 2.2. Signs of presence in the code
   - 2.3. Mitigations
   - 2.4. Safer alternatives
3. Pointers and Strings
4. Numeric errors
5. Exceptions
6. Injection Attacks
   - 6.1. Format string attacks
   - 6.2. Command injection
   - 6.3. SQL injection
   - 6.4. XML injection
7. Web Attacks
   - 7.1. Cross-site scripting (XSS)
   - 7.2. Cross-site request forgery (CSRF)
   - 7.3. Session hijacking
   - 7.4. Open redirect
8. Background on Automated Assessment Tools
9. The SWAMP
   - 9.1. What is the SWAMP?
   - 9.2. Using the SWAMP

Presenters' Short CV's

Barton Miller is the Vilas Distinguished Achievement and the Amar & Belinder Sohi Professor of Computer Science at the University of Wisconsin-Madison. He is Chief Scientist for the DHS Software Assurance Marketplace research facility and is Software Assurance Lead on the NSF Cybersecurity Center of Excellence. In addition, he co-directs the MIST software vulnerability assessment project in collaboration with his colleagues at the Autonomous University of Barcelona. He also leads the Paradyn Parallel Performance Tool project, which is investigating performance and instrumentation technologies for parallel and distributed applications and systems. His research interests include systems security, binary and malicious code analysis and instrumentation extreme scale systems, parallel and distributed program measurement and debugging, and mobile computing. Miller's research is supported by the U.S. Dept. of Homeland Security, U.S. Dept. of Energy, National Science Foundation, NATO, and various corporations.

In 1988, Miller founded the field of Fuzz random software testing, which is the foundation of many security and software engineering disciplines. In 1992, Miller (working with his then-student, Prof. Jeffrey Hollingsworth), founded the field of dynamic binary code instrumentation and coined the term "dynamic instrumentation". Dynamic instrumentation forms the basis for his current efforts in malware analysis and instrumentation.

Education

Ph.D. in Computer Science, May 1984. University of California, Berkeley.

Professional Experience

2005–present:	Adjunct Research Staff, IDA Center for Computing Sciences, Bowie, MD.
1995–present:	Professor, Computer Sciences Department, University of Wisconsin.
8/2009–1/2009:	Visiting Professor, Universitat Autònoma de Barcelona.
8/2009–1/2009:	Visiting Scientist, Barcelona Supercomputer Center.
1/1995–6/1995:	Visiting Scholar, Electrical Engineering Dept, Stanford University.
7/1989:	Distinguished Visiting Prof., Univ. Technica Federico St, Maria, Valparasio, Chile.

Selected Publications

1. Xiaozhu Meng and Barton P. Miller, "Binary Code Multi-Author Identification in Multi-Toolchain Scenarios", *submitted for publication,* October 2017.
2. Joseph O. Eichenhofer, Elisa Heymann and Barton P. Miller, "In-Depth Software Vulnerability Assessment of Container Terminal Systems", *2nd NATO Conference on Cyber Security in the Maritime Domain*, Souda, Crete, Greece, September 2017.
3. James A. Kupsch, Barton P. Miller, Vamshi Basupalli, and Josef Burger, "From Continuous Integration to Continuous Assurance", *IEEE Software Technology Conference,* Gaithersburg, Maryland, September 2017.
4. Xiaozhu Meng, Barton P. Miller, and Kwang-Sung Jun, "Identifying Multiple Authors in a Binary Program", *22nd European Symposium on Research in Computer Security (ESORICS),* Oslo, Norway, September 2017. Lecture Notes in Computer Science (LNCS) 10493, Spring Verlag, DOI: 10.1007/978-3-319-66399-9_16.
5. Elisa Heymann, Barton P. Miller, Mohammed J. Alghazzawi and David Incertis, "Addressing the Cyber-Security of Maritime Shipping" *2016 European Transport Conference (ETC)*, Barcelona, Spain, October 2016.
6. Xiaozhu Meng and Barton P. Miller, "Binary Code is Not Easy", *International Symposium on Software Testing and Analysis (ISSTA)*, Saarbruecken, Germany, July 2015.
7. James A. Kupsch, Elisa Heymann, Barton P. Miller and Vamshi Basupalli, "Bad and Good News about Using Software Assurance Tools", *Software: Practice and Experience*, April 2016. DOI: 10.1002/spe.2401.

8. Maxime Frydman, Guifre Ruiz, Elisa Heymann, Eduardo Cesar and Barton P. Miller, "Automating Risk Analysis of Software Design Models", *The Scientific World Journal* **2014**, Article ID 805856, June 2014. http://dx.doi.org/10.1155/2014/805856
9. E.R. Jacobson, A.R. Bernat, W.R. Williams, and B.P. Miller, "Detecting Code Reuse Attacks with a Model of Conformant Program Execution", *International Symposium on Engineering Secure Software and Systems (ESSOS)*, Munich, Germany, Feb 2014.
10. Xiaozhu Meng, Barton P. Miller, William R. Williams, Andrew R. Bernat, "Mining Software Repositories for Accurate Authorship", *29th IEEE International Conference on Software Maintenance,* Eindhoven, Netherlands, September 2013.

Selected Activities

Fellow, ACM
2011 R&D 100 Award
Vilas Fellow, University of Wisconsin, 2010-2011

Boards and Advisory Committees:

Advisory Board, DHS Application Security Threat Attack Modeling (ASTAM) project, 2017-present.
University of Wisconsin Ad Hoc Committee on Classified Research, 2014-present.
IDA Center for Computing Sciences, Program Review Committee (Chair), 2005-present.
Electronic Crimes Task Force (Chicago Area), U.S. Secret Service, 2002-2004.
Los Alamos Nat'l Lab Computing, Comm. and Networking Div. Review Comm, 2001-2007.
Univ of Wisc Center for Human Perf & Risk Analysis, Internal Advisory Board, 2002-2004.
Euro-Par Conference Advisory Board, 2002-present.
IEEE Technical Committee on Parallel Processing, 1995-present.
Advisory Committee, Tuskegee University High Perf Computing Program, 1996-2000.
Advisory Board, Int'l Summer Inst. on Parallel Comp. Arch., Languages, and Algs., Prague, July 1993.

Conference Committees:

Co-Chair, 2nd Shonan Meeting on Grid and Cloud Security: A Confluence, Japan, March 2014.
Program Co-Chair (Systems Software), International Conference on Supercomputing, Munich, Germany, May 2014.
Co-Chair, Shonan Meeting on Grid and Cloud Security: A Confluence, Japan, October 2012.
Co-Chair, Dagstuhl Seminar on Program Development for Extreme-Scale Computing, May 2010.
Chair, SC|2008 Technical Program – Tutorials, November 2008.
Steering Comm, DOE Software Development Tools for Petascale Computing, August 2007.
Co-Chair, Dagstuhl Sem. on Code Instrumentation & Modeling for Parallel Perf. Analysis, October 2007.
Performance Area Chair, Technical Papers Committee, SC|06, November 2006.

Outside Talks:

Distinguished Lecturer, Marquette University, Milwaukee, October 2016.
Invited Speaker, USENIX Security, Washington, DC, August 2015;.
Distinguished Lecturer, IBM T.J. Watson Research Center, June 2011.
Keynote Speaker, Open Grid Forum 28, Munich Germany, March 2010.
Distinguished Lecturer, College of Computing, Georgia Tech, Atlanta, April 2007.
Distinguished Lecturer, Triangle Computer Science Lecture Series (joint University of North Carolina, North Carolina State and Duke University), Raleigh, October 2006.
Plenary Speaker, Symposium on Adv. Computing Systems and Infrastructures, Tsukuba City, Japan, May 2005
Distinguished Lecturer, IBM T.J. Watson Research Center, December 2004.

Elisa Heymann is a Senior Scientist on the NSF Cybersecurity Center of Excellence at the University of Wisconsin-Madison, and an Associate Professor at the Autonomous University of Barcelona. She co-directs the MIST software vulnerability assessment at the Autonomous University of Barcelona, Spain.

She was also in charge of the Grid/Cloud security group at the UAB, and participated in two major Grid European Projects: EGI-InSPIRE and European Middleware Initiative (EMI). Heymann's research interests include security and resource management for Grid and Cloud environments. Her research is supported by the NSF, Spanish government, the European Commission, and NATO.

Education

Ph.D. in Computer Science, 2001. Universitat Autònoma de Barcelona (Spain).

Professional Experience

7/2016-present :	Senior Scientist, University of Wisconsin Madison.
2002–present:	Associate Professor, Universitat Autonoma de Barcelona, Spain.
1/2010-8/2010:	Visiting Professor, University of Wisconsin Madison.
9/1992-1/2002:	Assistant Professor, Universitat Autonoma de Barcelona, Spain.
1/1992–9/1992:	Research and Development, Fundación Instituto de Ingeniería, Venezuela.

Selected Publications

1. Joseph O. Eichenhofer, Elisa Heymann and Barton P. Miller, "In-Depth Software Vulnerability Assessment of Container Terminal Systems", 2nd NATO Conference on Cyber Security in the Maritime Domain, Souda, Crete, Greece, September 2017.
2. Elisa Heymann, Barton P. Miller, Mohammed J. Alghazzawi and David Incertis, "Addressing the Cyber-Security of Maritime Shipping" *2016 European Transport Conference (ETC),* Barcelona, Spain, October 2016.
3. James A. Kupsch, Elisa Heymann, Barton P. Miller and Vamshi Basupalli, "Bad and Good News about Using Software Assurance Tools", *Software: Practice and Experience,* April 2016. DOI: 10.1002/spe.2401.
4. Maxime Frydman, Guifre Ruiz, Elisa Heymann, Eduardo Cesar and Barton P. Miller, "Automating Risk Analysis of Software Design Models*", The Scientific World Journal* 2014 (2014), Article ID 805856, June 2014. http://dx.doi.org/10.1155/2014/805856. 2014.
5. Jairo D. Serrano, Elisa Heymann, Eduardo Cesar, Barton P. Miller, “Increasing Automated Vulnerability Assessment Accuracy on Cloud and Grid Middleware”, *9th International Conference on Information Security Practice and Experience (ISPEC),* Lanzhou, China, pp. 278-294, May 2013.
6. J. Serrano, E. Heymann, E. Cesar, and B. Miller, “Vulnerability Assessment Enhancement for Middleware", *Computing and Informatics Journal*, ISSN: 1335-9150, Vol. 31, No. 1, pp. 103-118. 2012.

Selected Activities

Co-Chair, 2nd Shonan Meeting on Grid and Cloud Security: A Confluence, Japan, March 2014. Co-Chair, Shonan Meeting on Grid and Cloud Security: A Confluence, Japan, October 2012. Co-organizer, European Condor Week. Barcelona, Spain, October 2008.

Reviewer for various conferences, including Supercomputing, EuroPar, and ICS 2014.

Reviewer of project proposals for the European Commission (FP6, FP7, and H2020): 2006-

present. Reviewer of project proposals for the Catalan Funding Agency (AGAUR), Barcelona, Spain: 2009-present. Reviewer of project proposals for the Valencian Funding Agency (AVAP), Valencia, Spain: 2010-present.