Diego Zamboni
Christopher Kruegel (Eds.)

LNCS 4219

Recent Advances in Intrusion Detection

9th International Symposium, RAID 2006
Hamburg, Germany, September 2006
Proceedings

Springer

Diego Zamboni Christopher Kruegel (Eds.)

Recent Advances in Intrusion Detection

9th International Symposium, RAID 2006
Hamburg, Germany, September 20-22, 2006
Proceedings

Springer

Volume Editors

Diego Zamboni
IBM Research GmbH
Zurich Research Laboratory
Säumerstr. 4, Postfach, 8803 Rüschlikon, Switzerland
E-mail: dza@zurich.ibm.com

Christopher Kruegel
Technical University of Vienna
Secure Systems Lab
Treitlstrasse 3, A-1040 Vienna, Austria
E-mail: chris@auto.tuwien.ac.at

Library of Congress Control Number: 2006932117

CR Subject Classification (1998): K.6.5, K.4, E.3, C.2, D.4.6

LNCS Sublibrary: SL 4 – Security and Cryptology

ISSN 0302-9743
ISBN-10 3-540-39723-X Springer Berlin Heidelberg New York
ISBN-13 978-3-540-39723-6 Springer Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11856214 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 4219

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

Lecture Notes in Computer Science

For information about Vols. 1–4087

please contact your bookseller or Springer

Vol. 4228: D.E. Lightfoot, C.A. Szyperski (Eds.), Modular Programming Languages. X, 415 pages. 2006.

Vol. 4219: D. Zamboni, C. Kruegel (Eds.), Recent Advances in Intrusion Detection. XII, 331 pages. 2006.

Vol. 4208: M. Gerndt, D. Kranzlmüller (Eds.), High Performance Computing and Communications. XXII, 938 pages. 2006.

Vol. 4206: P. Dourish, A. Friday (Eds.), UbiComp 2006: Ubiquitous Computing. XIX, 526 pages. 2006.

Vol. 4193: T.P. Runarsson, H.-G. Beyer, E. Burke, J.J. Merelo-Guervós, L. D. Whitley, X. Yao (Eds.), Parallel Problem Solving from Nature - PPSN IX. XIX, 1061 pages. 2006.

Vol. 4192: B. Mohr, J.L. Träff, J. Worringen, J. Dongarra (Eds.), Recent Advances in Parallel Virtual Machine and Message Passing Interface. XVI, 414 pages. 2006.

Vol. 4188: P. Sojka, I. Kopeček, K. Pala (Eds.), Text, Speech and Dialogue. XIV, 721 pages. 2006. (Sublibrary LNAI).

Vol. 4187: J.J. Alferes, J. Bailey, W. May, U. Schwertel (Eds.), Principles and Practice of Semantic Web Reasoning. XI, 277 pages. 2006.

Vol. 4186: C. Jesshope, C. Egan (Eds.), Advances in Computer Systems Architecture. XIV, 605 pages. 2006.

Vol. 4185: R. Mizoguchi, Z. Shi, F. Giunchiglia (Eds.), The Semantic Web – ASWC 2006. XX, 778 pages. 2006.

Vol. 4184: M. Bravetti, M. Núñez, G. Zavattaro (Eds.), Web Services and Formal Methods. X, 289 pages. 2006.

Vol. 4183: J. Euzenat, J. Domingue (Eds.), Artificial Intelligence: Methodology, Systems, and Applications. XIII, 291 pages. 2006. (Sublibrary LNAI).

Vol. 4180: M. Kohlhase, OMDoc – An Open Markup Format for Mathematical Documents [version 1.2]. XIX, 428 pages. 2006. (Sublibrary LNAI).

Vol. 4178: A. Corradini, H. Ehrig, U. Montanari, L. Ribeiro, G. Rozenberg (Eds.), Graph Transformations. XII, 473 pages. 2006.

Vol. 4176: S.K. Katsikas, J. Lopez, M. Backes, S. Gritzalis, B. Preneel (Eds.), Information Security. XIV, 548 pages. 2006.

Vol. 4175: P. Bücher, B.M.E. Moret (Eds.), Algorithms in Bioinformatics. XII, 402 pages. 2006. (Sublibrary LNBI).

Vol. 4174: K. Franke, K.-R. Müller, B. Nickolay, R. Schäfer (Eds.), Pattern Recognition. XX, 773 pages. 2006.

Vol. 4169: H.L. Bodlaender, M.A. Langston (Eds.), Parameterized and Exact Computation. XI, 279 pages. 2006.

Vol. 4168: Y. Azar, T. Erlebach (Eds.), Algorithms – ESA 2006. XVIII, 843 pages. 2006.

Vol. 4165: W. Jonker, M. Petković (Eds.), Secure, Data Management. X, 185 pages. 2006.

Vol. 4163: H. Bersini, J. Carneiro (Eds.), Artificial Immune Systems. XII, 460 pages. 2006.

Vol. 4162: R. Královič, P. Urzyczyn (Eds.), Mathematical Foundations of Computer Science 2006. XV, 814 pages. 2006.

Vol. 4160: M. Fisher, W.v.d. Hoek, B. Konev, A. Lisitsa (Eds.), Logics in Artificial Intelligence. XII, 516 pages. 2006. (Sublibrary LNAI).

Vol. 4159: J. Ma, H. Jin, L.T. Yang, J.J.-P. Tsai (Eds.), Ubiquitous Intelligence and Computing. XXII, 1190 pages. 2006.

Vol. 4158: L.T. Yang, H. Jin, J. Ma, T. Ungerer (Eds.), Autonomic and Trusted Computing. XIV, 613 pages. 2006.

Vol. 4156: S. Amer-Yahia, Z. Bellahsène, E. Hunt, R. Unland, J.X. Yu (Eds.), Database and XML Technologies. IX, 123 pages. 2006.

Vol. 4155: O. Stock, M. Schaerf (Eds.), Reasoning, Action and Interaction in AI Theories and Systems. XVIII, 343 pages. 2006. (Sublibrary LNAI).

Vol. 4153: N. Zheng, X. Jiang, X. Lan (Eds.), Advances in Machine Vision, Image Processing, and Pattern Analysis. XIII, 506 pages. 2006.

Vol. 4152: Y. Manolopoulos, J. Pokorný, T. Sellis (Eds.), Advances in Databases and Information Systems. XV, 448 pages. 2006.

Vol. 4151: A. Iglesias, N. Takayama (Eds.), Mathematical Software - ICMS 2006. XVII, 452 pages. 2006.

Vol. 4150: M. Dorigo, L.M. Gambardella, M. Birattari, A. Martinoli, R. Poli, T. Stützle (Eds.), Ant Colony Optimization and Swarm Intelligence. XVI, 526 pages. 2006.

Vol. 4149: M. Klusch, M. Rovatsos, T.R. Payne (Eds.), Cooperative Information Agents X. XII, 477 pages. 2006. (Sublibrary LNAI).

Vol. 4148: J. Vounckx, N. Azemard, P. Maurine (Eds.), Integrated Circuit and System Design. XVI, 677 pages. 2006.

Vol. 4146: J.C. Rajapakse, L. Wong, R. Acharya (Eds.), Pattern Recognition in Bioinformatics. XIV, 186 pages. 2006. (Sublibrary LNBI).

Vol. 4144: T. Ball, R.B. Jones (Eds.), Computer Aided Verification. XV, 564 pages. 2006.

Vol. 4139: T. Salakoski, F. Ginter, S. Pyysalo, T. Pahikkala, Advances in Natural Language Processing. XVI, 771 pages. 2006. (Sublibrary LNAI).

Vol. 4138: X. Cheng, W. Li, T. Znati (Eds.), Wireless Algorithms, Systems, and Applications. XVI, 709 pages. 2006.

Vol. 4137: C. Baier, H. Hermanns (Eds.), CONCUR 2006 – Concurrency Theory. XIII, 525 pages. 2006.

Vol. 4136: R.A. Schmidt (Ed.), Relations and Kleene Algebra in Computer Science. XI, 433 pages. 2006.

Vol. 4135: C.S. Calude, M.J. Dinneen, G. Păun, G. Rozenberg, S. Stepney (Eds.), Unconventional Computation. X, 267 pages. 2006.

Vol. 4134: K. Yi (Ed.), Static Analysis. XIII, 443 pages. 2006.

Vol. 4133: J. Gratch, M. Young, R. Aylett, D. Ballin, P. Olivier (Eds.), Intelligent Virtual Agents. XIV, 472 pages. 2006. (Sublibrary LNAI).

Vol. 4132: S. Kollias, A. Stafylopatis, W. Duch, E. Oja (Eds.), Artificial Neural Networks – ICANN 2006, Part II. XXXIV, 1028 pages. 2006.

Vol. 4131: S. Kollias, A. Stafylopatis, W. Duch, E. Oja (Eds.), Artificial Neural Networks – ICANN 2006, Part I. XXXIV, 1008 pages. 2006.

Vol. 4130: U. Furbach, N. Shankar (Eds.), Automated Reasoning. XV, 680 pages. 2006. (Sublibrary LNAI).

Vol. 4129: D. McGookin, S. Brewster (Eds.), Haptic and Audio Interaction Design. XII, 167 pages. 2006.

Vol. 4128: W.E. Nagel, W.V. Walter, W. Lehner (Eds.), Euro-Par 2006 Parallel Processing. XXXIII, 1221 pages. 2006.

Vol. 4127: E. Damiani, P. Liu (Eds.), Data and Applications Security XX. X, 319 pages. 2006.

Vol. 4126: P. Barahona, F. Bry, E. Franconi, N. Henze, U. Sattler, Reasoning Web. X, 269 pages. 2006.

Vol. 4124: H. de Meer, J.P. G. Sterbenz (Eds.), Self-Organizing Systems. XIV, 261 pages. 2006.

Vol. 4121: A. Biere, C.P. Gomes (Eds.), Theory and Applications of Satisfiability Testing - SAT 2006. XII, 438 pages. 2006.

Vol. 4120: J. Calmet, T. Ida, D. Wang (Eds.), Artificial Intelligence and Symbolic Computation. XIII, 269 pages. 2006. (Sublibrary LNAI).

Vol. 4119: C. Dony, J.L. Knudsen, A. Romanovsky, A. Tripathi (Eds.), Advanced Topics in Exception Handling Components. X, 302 pages. 2006.

Vol. 4117: C. Dwork (Ed.), Advances in Cryptology - CRYPTO 2006. XIII, 621 pages. 2006.

Vol. 4116: R. De Prisco, M. Yung (Eds.), Security and Cryptography for Networks. XI, 366 pages. 2006.

Vol. 4115: D.-S. Huang, K. Li, G.W. Irwin (Eds.), Computational Intelligence and Bioinformatics, Part III. XXI, 803 pages. 2006. (Sublibrary LNBI).

Vol. 4114: D.-S. Huang, K. Li, G.W. Irwin (Eds.), Computational Intelligence, Part II. XXVII, 1337 pages. 2006. (Sublibrary LNAI).

Vol. 4113: D.-S. Huang, K. Li, G.W. Irwin (Eds.), Intelligent Computing, Part I. XXVII, 1331 pages. 2006.

Vol. 4112: D.Z. Chen, D. T. Lee (Eds.), Computing and Combinatorics. XIV, 528 pages. 2006.

Vol. 4111: F.S. de Boer, M.M. Bonsangue, S. Graf, W.-P. de Roever (Eds.), Formal Methods for Components and Objects. VIII, 447 pages. 2006.

Vol. 4110: J. Díaz, K. Jansen, J.D.P. Rolim, U. Zwick (Eds.), Approximation, Randomization, and Combinatorial Optimization. XII, 522 pages. 2006.

Vol. 4109: D.-Y. Yeung, J.T. Kwok, A. Fred, F. Roli, D. de Ridder (Eds.), Structural, Syntactic, and Statistical Pattern Recognition. XXI, 939 pages. 2006.

Vol. 4108: J.M. Borwein, W.M. Farmer (Eds.), Mathematical Knowledge Management. VIII, 295 pages. 2006. (Sublibrary LNAI).

Vol. 4106: T.R. Roth-Berghofer, M.H. Göker, H. A. Güvenir (Eds.), Advances in Case-Based Reasoning. XIV, 566 pages. 2006. (Sublibrary LNAI).

Vol. 4105: B. Gunsel, A.K. Jain, A. M. Tekalp, B. Sankur (Eds.), Multimedia, Content Representation, Classification and Security. XIX, 804 pages. 2006.

Vol. 4104: T. Kunz, S.S. Ravi (Eds.), Ad-Hoc, Mobile, and Wireless Networks. XII, 474 pages. 2006.

Vol. 4103: J. Eder, S. Dustdar (Eds.), Business Process Management Workshops. XI, 508 pages. 2006.

Vol. 4102: S. Dustdar, J.L. Fiadeiro, A. Sheth (Eds.), Business Process Management. XV, 486 pages. 2006.

Vol. 4099: Q. Yang, G. Webb (Eds.), PRICAI 2006: Trends in Artificial Intelligence. XXVIII, 1263 pages. 2006. (Sublibrary LNAI).

Vol. 4098: F. Pfenning (Ed.), Term Rewriting and Applications. XIII, 415 pages. 2006.

Vol. 4097: X. Zhou, O. Sokolsky, L. Yan, E.-S. Jung, Z. Shao, Y. Mu, D.C. Lee, D. Kim, Y.-S. Jeong, C.-Z. Xu (Eds.), Emerging Directions in Embedded and Ubiquitous Computing. XXVII, 1034 pages. 2006.

Vol. 4096: E. Sha, S.-K. Han, C.-Z. Xu, M.H. Kim, L.T. Yang, B. Xiao (Eds.), Embedded and Ubiquitous Computing. XXIV, 1170 pages. 2006.

Vol. 4095: S. Nolfi, G. Baldassarre, R. Calabretta, J.C. T. Hallam, D. Marocco, J.-A. Meyer, O. Miglino, D. Parisi (Eds.), From Animals to Animats 9. XV, 869 pages. 2006. (Sublibrary LNAI).

Vol. 4094: O. H. Ibarra, H.-C. Yen (Eds.), Implementation and Application of Automata. XIII, 291 pages. 2006.

Vol. 4093: X. Li, O.R. Zaïane, Z. Li (Eds.), Advanced Data Mining and Applications. XXI, 1110 pages. 2006. (Sublibrary LNAI).

Vol. 4092: J. Lang, F. Lin, J. Wang (Eds.), Knowledge Science, Engineering and Management. XV, 664 pages. 2006. (Sublibrary LNAI).

Vol. 4091: G.-Z. Yang, T. Jiang, D. Shen, L. Gu, J. Yang (Eds.), Medical Imaging and Augmented Reality. XIII, 399 pages. 2006.

Vol. 4090: S. Spaccapietra, K. Aberer, P. Cudré-Mauroux (Eds.), Journal on Data Semantics VI. XI, 211 pages. 2006.

Vol. 4089: W. Löwe, M. Südholt (Eds.), Software Composition. X, 339 pages. 2006.

Vol. 4088: Z.-Z. Shi, R. Sadananda (Eds.), Agent Computing and Multi-Agent Systems. XVII, 827 pages. 2006. (Sublibrary LNAI).

¥410.00元

Preface

On behalf of the Program Committee, it is our pleasure to present the proceedings of the 9th Symposium on Recent Advances in Intrusion Detection (RAID 2006), which took place in Hamburg, Germany, on September 20-22, 2006.

As every year since 1998, the symposium brought together leading researchers and practitioners from academia, government and industry to discuss intrusion detection research and practice. We had sessions on anomaly and specification-based detection, network-based intrusion detection, attacks against intrusion detection systems, IDS evaluation and malware analysis.

The RAID 2005 Program Committee received 93 paper submissions from all over the world, including 15 papers submitted as "Big Challenge, Big Idea" papers. All the submissions were carefully reviewed by several members of the Program Committee and evaluated on the basis of scientific novelty, importance to the field, and technical quality. Final selection took place at the Program Committee meeting held on June 1st and 2nd in Zürich, Switzerland. Sixteen papers were selected for presentation and publication in the conference proceedings, placing RAID among the most competitive conferences in the area of computer security.

This year we announced "Big Challenge, Big Idea" as a theme. We encouraged submissions in a separate category, looking for papers that described fundamental problems that have not yet been tackled by intrusion detection research, or bold, risky or controversial ideas for potential research or solutions.

A successful symposium is the result of the joint effort of many people. In particular, we would like to thank all the authors who submitted papers, whether accepted or not. We also thank the Program Committee members and additional reviewers for their hard work in evaluating the submissions. In addition, we want to thank the General Chair, Dieter Gollmann, for handling the conference arrangements, Robert Cunningham for publicizing the conference, James Riordan for putting together the conference proceedings, Klaus-Peter Kossakowski for finding sponsor support, and Jan Meier for maintaining the conference Web site. Finally, we extend our thanks to the Northwest Security Institute (NSWI) and Cisco Systems for their sponsorship of student scholarships.

September 2006

Diego Zamboni
Christopher Kruegel

Organization

RAID 2006 was organized by the Technical University of Hamburg-Harburg and held in conjunction with ESORICS 2006.

Conference Chairs

General Chairs	Dieter Gollmann (Technical University Hamburg-Harburg), Andreas Günter(HiTech)
Program Chair	Diego Zamboni (IBM Zurich Research Laboratory)
Program Co-chair	Christopher Kruegel (Technical University Vienna)
Publication Chair	James Riordan (IBM Zurich Research Laboratory)
Publicity Chair	Robert Cunningham (MIT Lincoln Laboratory)
Sponsorship Chair	Klaus-Peter Kossakowski (PRESECURE Consulting)

Program Committee

Magnus Almgren	Chalmers University, Sweden
Michael Behringer	Cisco Systems, Inc., USA
Sungdeok Cha	Korea Advanced Institute of Science and Technology, Korea
Steve J. Chapin	Systems Assurance Institute, Syracuse University, USA
Andrew Clark	Queensland University of Technology, Australia
Crispin Cowan	Novell, USA
Robert Cunningham	MIT Lincoln Laboratory, USA
Olivier De Vel	Department of Defence, Australia
Farnam Jahanian	University of Michigan and Arbor Networks, USA
Somesh Jha	University of Wisconsin, Madison, USA
Klaus-Peter Kossakowski	DFN-CERT, Germany
Christopher Kruegel	Technical University Vienna, Austria
Kwok-Yan Lam	Tsinghua University, China
Ulf Lindqvist	SRI International, USA
Raffael Marty	ArcSight, Inc., USA
George Mohay	Queensland University of Technology, Australia
Benjamin Morin	Supélec, France

Program Committee (Continued)

Peng Ning	North Carolina State University, USA
James Riordan	IBM Zurich Research Laboratory, Switzerland
Rei Safavi-Naini	University of Wollongong, Australia
Dawn Song	Carnegie Mellon University, USA
Sal Stolfo	Department of Computer Science, Columbia University, USA
Toshihiro Tabata	Okayama University, Japan
Kymie Tan	Carnegie Mellon University, USA
Vijay Varadharajan	Macquarie University, Australia
Giovanni Vigna	University of California at Santa Barbara, USA
Jianying Zhou	Institute for Infocomm Research, Singapore

Steering Committee

Marc Dacier (chair)	Eurecom, France
Hervé Debar	France Telecom R&D, France
Deborah Frincke	Pacific Northwest National Lab, USA
Ming-Yuh Huang	The Boeing Company, USA
Erland Jonsson	Chalmers, Sweden
Wenke Lee	Georgia Institute of Technology, USA
Ludovic Mé	Supélec, France
S. Felix Wu	UC Davis, USA
Andreas Wespi	IBM Research, Switzerland
Alfonso Valdes	SRI International, USA
Giovanni Vigna	UCSB, USA

Additional Reviewers

Hirotake Abe	Japan Science and Technology Agency, Japan
Stig Andersson	Queensland University of Technology, Australia
Mark Branagan	Queensland University of Technology, Australia
Hyung Chan Kim	Gwangju Institute of Science and Technology, Korea
Malcolm Corney	Queensland University of Technology, Australia
Siu-Leung Chung	Open University of Hong Kong
Gabriela F. Cretu	CS Department Columbia University, USA
Meng Ge	Tsinghua University, China
Daniel Hedin	Chalmers University of Technology and Göteborg University, Sweden

Additional Reviewers (Continued)

Matt Henricksen	Queensland University of Technology, Australia
Jeffrey Horton	University of Wollongong, Australia
Corrado Leita	Eurecom, France
Wei-Jen Li	CS Department Columbia University, USA
Zhuowei Li	Indiana University, USA
Liang Lu	University of Wollongong, Australia
Andreas Moser	Technical University Vienna
Yoshihiro Oyama	University of Electro-Communications, Japan
Janak Parekh	CS Department, Columbia University, USA
Van Hau Pham.	Eurecom, France
Bradley Schatz	Queensland University of Technology, Australia
Jinyang Shi	Tsinghua University, China
Hongwei Sun	Tsinghua University, China
Olivier Thonnard	Eurecom, France
Uday K. Tupakula	Macquarie University, Australia
Ke Wang	CS Department, Columbia University, USA
Jacob Zimmermann	Queensland University of Technology, Australia

Table of Contents

Recent Advances in Intrusion Detection

Anomaly Detection

Attacks

System Evaluation and Threat Assessment

Malware Collection and Analysis

Anomaly- and Specification-Based Detection

Network Intrusion Detection

A Framework for the Application of Association Rule Mining in Large Intrusion Detection Infrastructures

James J. Treinen[1] and Ramakrishna Thurimella[2]

[1] IBM Global Services, Boulder, CO 80301, USA
jamestr@us.ibm.com
[2] University of Denver, Denver, CO 80208, USA
ramki@cs.du.edu

Abstract. The high number of false positive alarms that are generated in large intrusion detection infrastructures makes it difficult for operations staff to separate false alerts from real attacks. One means of reducing this problem is the use of meta alarms, or rules, which identify known attack patterns in alarm streams. The obvious risk with this approach is that the rule base may not be complete with respect to every true attack profile, especially those which are new. Currently, new rules are discovered manually, a process which is both costly and error prone. We present a novel approach using association rule mining to shorten the time that elapses from the appearance of a new attack profile in the data to its definition as a rule in the production monitoring infrastructure.

Keywords: Association Rules, Data Mining, Intrusion Detection, Graph Algorithms.

1 Introduction

Attempts to compromise networked computing resources generally consist of multiple steps. The first of these is the reconnaissance phase, consisting of the identification of target operating systems, port scanning, and vulnerability enumeration. This is followed by the exploitation of the weaknesses discovered during the initial intelligence gathering process. A successful attack often ends with the installation of back door channels so that the attacker can easily gain access to the system in the future [29].

If an intrusion detection infrastructure is in use at the victim network during this process, each action by the attacker has the potential to raise an alarm, alerting the security staff to the presence of malicious activity in the network. Generally speaking, intrusion detection sensors do not have the ability to aggregate the alarms for the discrete activities into an end-to-end attack profile. Given that an alarm is raised for each perceived malicious action, the typical intrusion detection sensor can generate many thousands of alarms per day. Unfortunately, the vast majority of these alarms are false positives [20], and the task of separating the real attacks from false alarms quickly becomes daunting.

D. Zamboni and C. Kruegel (Eds.): RAID 2006, LNCS 4219, pp. 1–18, 2006.

As noted by Lippmann, et al. in [26], the deployment of an inaccurate Intrusion Detection Sensor (IDS) can have undesirable effects in addition to simply missing certain types of attacks. The first of these is the potential to reduce the level of vigilant monitoring by security operations staff, due to the false sense of security provided by the IDS. Secondly, using operations staff to examine all of the alarms produced in a day can make the deployment of a typical IDS system extremely expensive in terms of support and labor costs. These issues are further compounded in large monitoring infrastructures where the number of managed sensors can easily reach into the thousands, generating millions of alerts per day.

The context for our experiments is that of a *large* Managed Security Service Provider (MSSP). Our experiments were conducted on a production data set that was generated by roughly 1000 IDS sensors. The sensor technologies used to generate the data set represented multiple vendors and versions of their software, and were installed across 135 distinct customer networks. The alarm logs generated by the sensors were consolidated at a Security Operations Center (SOC) which used a third party Enterprise Security Manager (ESM) with the ability to monitor the incoming alarm stream and match the alarms against a predefined set of meta rules. It is these meta rules which the operations staff use to detect intrusions across the networks they monitor. Similar to signature based intrusion detection sensors, the ESM uses pattern matching to detect predefined patterns in the incoming alarm streams. If the base alarms arriving at the ESM consolidation point match a predefined attack rule in the monitoring engine, a meta alarm is triggered and displayed on the operations staff's console for inspection.

Because new vulnerabilities are discovered every day, new alarm signatures are continuously installed on the intrusion detection sensors. This highly dynamic environment produces a genuine challenge in terms of keeping the rule base in the ESM current. Our framework provides a means of reducing the amount of labor required to keep the rules current in the ESM, while at the same time significantly reducing the amount of time which elapses from the appearance of a new attack profile in the data to installation of the corresponding rule in the production monitoring environment.

The time from the appearance of new attack profiles to the time when new rules describing them are implemented is *critical*. Any delay in updating the rule base could result in potentially undetected attacks. The amount of manual inspection currently required to discover new rules makes staffing to meet these time demands very expensive. We have found that using our framework to automate this task drastically decreases the amount of manual inspection required. This in turn has the net effect of decreasing the time from discovery to implementation as well as decreasing the over all cost of maintenance.

The concept of association rule mining for intrusion detection was introduced by Lee, et al. in [22], and is extended in [6,24,27]. Their approach is to use the rules returned by the association rule algorithm to prove that causal relationships exist between a user, and the type of entries that are logged in the audit

data as a result of their actions on the system. Our research has shown that in the same manner that [22,24] were able to demonstrate the existence of causal relationships between users and the entries logged in system audit data as a result of their actions, it is possible to show causal relationships between an attacker and the combination of alarms which are generated in intrusion detection logs as a result of their behavior in a network. We were then able to use the patterns which were discovered using our data mining technique to configure new rules for the ESM system in a rapid and economical way. As a means of demonstrating this, we include examples of attack activity which answer the following questions:

1. What techniques did the attacker employ?
2. How were these techniques manifested as patterns in the IDS alarm logs?
3. Was our framework able to detect these patterns?
4. How did the discovered patterns result in a new rule in the ESM?

As with all data mining solutions, much up-front work must be done adjusting the parameters for the algorithm so that optimal results are obtained. There is no silver bullet configuration, and it is noted throughout the literature that when using association rule mining, the features which are chosen for examination are critical to the success of the algorithm [24,30].

The remainder of this paper is organized as follows. Related work is discussed in Section 2. Section 3 provides an overview of the experimental environment, a brief description of data mining terminology, and a discussion of representing alarms as directed graphs. Section 4 defines our approach, including a novel alarm filtering technique. Section 5 describes our results, and provides example rules which were generated using our framework. Section 6 presents concluding remarks.

2 Related Work

Many data mining techniques have been applied to intrusion detection. The vast majority of the research has concentrated on mining various types of system audit data, or raw network traffic in order to build more accurate IDS devices [6,13,22,23,24,25,30,33,34,35].

The use of data mining has also been employed to examine alarm logs, specifically using cluster analysis to classify alarms into attack and benign categories [20,24] and to perform root cause analysis regarding the cause of false alarms in [17,18,20,21]. The results obtained using cluster analysis can vary widely depending on which algorithm and distance measure is used. These issues are discussed at length in [10,14,20,22,24,30,33,37].

In order to be truly effective, the use of data mining techniques must be one step in an over all Knowledge Discovery in Databases (KDD) process. This case is made repeatedly in the literature, e.g. [30] who use cluster analysis solely as the initial step in their data exploration. It is reiterated in [17,18,20,21] that although

the research tends to focus on the mining algorithm employed, it is only one step in the overall KDD process. They also note that without all of these steps, data mining runs a high risk of finding meaningless or uninteresting patterns. It is for this reason that [37] propose their end-to-end KDD architecture. Julisch outlines the basic KDD steps as follows in [18], as condensed from their original definition in [9] :

1. Understand the application domain
2. Data integration and selection
3. Data mining
4. Pattern evaluation
5. Knowledge presentation

A similar outline is made in [30], who also note that once a group of domain experts is consulted, the entire process should be automated.

3 Preliminaries

3.1 Experimental Environment

Figure 1 describes our data mining architecture. As the alarms arrive at the SOC, they are stored temporarily in a database on the monitoring engine. From this database we extracted the set of all alarms generated in a single day for all networks and loaded them into a data warehouse. It is on this warehouse that we executed the data mining algorithms with the goal of generating new monitoring rules for installation in the ESM.

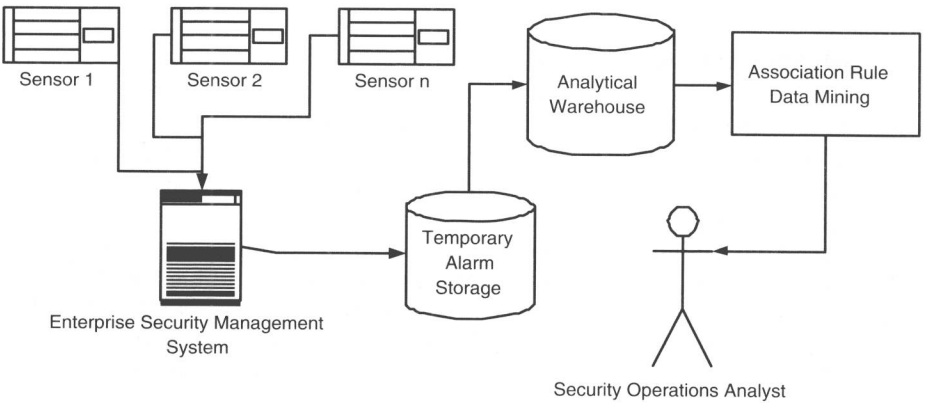

Fig. 1. The Association Rules Data Mining Architecture

3.2 Data Mining Terminology

In our analysis, we employ the use of association rule mining [1]. Because the field of data mining is very mature, rather than focusing on improving existing rule algorithms, we make use of the functionality that is available in DB2 Intelligent Miner for Modeling v8.2, which provides a fast algorithm for finding association rules. The main goal of association rule mining is to locate non-obvious interrelationships between members of a large data set [16]. The goal of our analysis is to find associations between the various attack signatures and IP addresses which constitute true attacks on the network, and capture them as rules in the ESM rule engine so that the SOC can easily detect future instances of the attack. The association rules algorithm generates rules in the following form, as well as some statistics which describe their strength and quality.

$$[x][y] \rightarrow [z]$$

$$Support = 50$$

$$Confidence = 80$$

This rule indicates that a relationship exists between the items x, y and z. Specifically, the rule states that whenever x and y were present in a given grouping, known as a transaction, then z was present as well. The Support value states that this specific grouping of three items represents 50 percent of the transactions which were examined. The Confidence value states that 80 percent of the time that the items x and y were found together, the item z was also found [16].

Formally, let $I = \{i_1, i_2, ..., i_n\}$ be a set of items. Given a set of transactions D, where each transaction is defined as a set of items $T \subseteq I$, a transaction T contains X if $X \subseteq T$. An association rule is an implication $X \Rightarrow Y$, where $X \subset I$, $Y \subset I$, and $X \cap Y = \emptyset$. The association rule $X \Rightarrow Y$ holds in the transaction set D with a Confidence c if c percent of transactions in D which contain X also contain Y. The association rule $X \Rightarrow Y$ has a Support value s in the transaction set D if s percent of the transactions in D contain $X \cup Y$ [1].

In our results, the Support values are typically less than 5 percent. This is due to the fact that thousands of signatures exist in the monitoring infrastructure, and generally the rules which are discovered cover only a small percentage of the total signature set for a given day.

3.3 Modeling Alarms as Directed Graphs

In order to facilitate a novel technique for filtering the number of alarms which must be analyzed during the mining process, we generated a directed graph which modeled the alarms to be examined. Each entry in the data warehouse included both the source IP address and destination IP address for which the alarm was raised. We deduced the direction of each potential attack from this information. We then generated a directed graph $G = (V, E)$ such that each IP address was represented as a vertex in the graph, and each edge was represented