REAL-TIME MULTISTAGE ATTACK AWARENESS THROUGH ENHANCED INTRUSION ALERT CLUSTERING *

Sunu Mathew, Daniel Britt, Richard Giomundo, Shambhu Upadhyaya
Department of Computer Science and Engineering
State University of New York at Buffalo
Buffalo, NY 14260
Email: (smathew2, djbritt, giomundo, shambhu)@cse.buffalo.edu

Moises Sudit, Adam Stotz
Department of Industrial Engineering
State University of New York at Buffalo
Buffalo, NY 14260
Email:(sudit, astotz)@eng.buffalo.edu

ABSTRACT

Correlation and fusion of intrusion alerts to provide effective Situation Awareness of cyber-attacks has become an active area of research. Snort is the most widely deployed intrusion detection sensor. For many networks and their system administrators, the alerts generated by Snort are the primary indicators of network misuse and attacker activity. However, the volume of the alerts generated in typical networks makes real-time attack scenario comprehension difficult. In this paper, we present an attack-stage oriented classification of alerts using Snort as an example, and demonstrate that this effectively improves real-time Situation Awareness of multistage attacks. We also incorporate this scheme into a real-time attack detection framework and prototype presented by the authors in previous work and provide some results from testing against multistage attack scenarios.

INTRODUCTION

Several schemes have been proposed for Intrusion Detection System (IDS) alert correlation including a framework for real-time attack scenario detection by the authors in [2]. The primary (and often, the only) source of alerts in most networks is the open source sensor Snort [11]. The volume of alerts generated by Snort on a typical network is of the order of thousands a day. The alert messages that Snort generates are cryptic and often exploit-specific. Thus, it is difficult for an analyst to derive a high-level view of attacker activity and attack progression that can enable him to take timely actions. The need therefore, is this: An effective situation awareness mechanism that can quickly indicate major stages or broad outlines of an attack scenario and provide useful information for network defense. In this paper, we present an alert categorization model that quickly and effectively groups alerts in a live alert stream to indicate relevant stages of a multistage attack scenario. This provides the analyst with the necessary information to be able to take time-critical decisions without being overwhelmed by the volume of alerts. This alert categorization method is incorporated into the alert fusion system presented by the authors in [2]. We demonstrate the utility of the approach in enhancing the output of the fusion engine and in providing improved Situation Awareness by testing with actual multistage attack scenarios.

The main contributions of this paper are:

- An attack stage based nomenclature for alert

*Research supported in part by Alion Science and Technology subcontract F30602-03-C-0245 from ARDA and AFRL programs

clustering and categorization aimed at enhancing Situation Awareness of attack scenarios

- Experimental results with multistage attacks that demonstrate the utility of our approach

The rest of this paper is organized as follows: Section 2 reviews related work, Section 3 presents our alert clustering or classification scheme and Section 4 presents experimental results. Section 5 relates our work to the military's Indications and Warning (I & W) framework and Section 6 concludes with some directions for future work.

BACKGROUND AND RELATED WORK

A detailed classification of Intrusion Detection Sensors is presented in [1]. Several schemes to classify computer attacks have also been proposed in the literature. In [5], computer attacks are classified on the basis of the attack techniques such as masquerade and hardware misuse. Subsequently in [6], the approach is extended to classify computer attacks based on multiple dimensions, such as both intrusion technique and intrusion result. In [3], a detailed computer security incident taxonomy is presented. Here an effort is made to define what constitutes an event, an attack and an incident. Various aspects of a computer security incident such as possible attackers, attack tools, vulnerabilities, results and objectives are considered. A target-centric approach to classifying computer attacks is presented in [4]. Here a detailed ontology for intrusion detection is presented in which the target, means, consequences and location of an attack are considered and appropriate taxonomies are suggested for these.

However, none of the above research efforts attempt to classify or relate intrusion events from the point of view of a *multistage* attack. (Some of the attack consequences that the above cited works define could however be used for this purpose). Situation Awareness of an evolving multistage cyber-attack is required for accurate Indications and Warning as defined by the military and intelligence communities ([7], [8]). This requires an analyst to be able to classify and relate individual attacks based on their role in reconnaissance, intrusion, escalation of privileges and finally, goal(s) that may be driving the attacker and his actions. Another fact is that although most other efforts introduce taxonomies, they fail to apply this scheme to actual intrusion alerts from specific sensors. Thus, previous efforts, though useful, are deficient when dealing with actual IDS alerts. The view that taxonomies (and thus the individual classes that they consist of) should be mutually exclusive ([9], [6]) and unambiguous is not supported when one examines alerts that are generated by real sensors, say Snort. As an example, consider Snort alert *NETBIOS SMB-DS DCERPC Remote Activation bind attempt* with Snort ID 2252. The result of this attack is either a *Denial-of-Service* or *Unauthorized remote administrative access*. These two results mean different things to the analyst trying to detect a coordinated attack against a high-value target network. Actual detection of what the attacker intends in this case is possible only by correlating this event with others. It must also be noted that Snort provides its own high-level classification of alerts in the form of rules files (e.g. web-attacks.rules, ftp.rules), but these only indicate the general category of the attack and not what the attacker achieves or the logical progression of a goal-oriented attack.

ALERT CLASSIFICATION SCHEME

Our alert classification scheme is designed to categorize intrusion detection sensor alerts into groups that most effectively indicate their *stage* in a multistage attack (Thus, this is different from the term clustering as used in Computer Science with respect to algorithms such as *K-Means clustering*). Effective situation awareness of cyber-attacks requires fusion of alerts from both host and network IDSs. Our scheme describes classes into which alerts can be mapped, using Snort alerts as clarifying examples (other sensor alerts could be similarly mapped). An alert can be part of multiple classes. Each class has a two part name - the first indicates the general category (e.g., *Reconnaissance, Intrusion, Privilege Escalation and Goal*), and the second indicates a specific sub-category. Alert descriptions are taken from the Snort signature database. Table 1 summarizes the classification scheme. Additional miscellaneous categories are introduced for alerts that may not indicate a specific attack stage, but are more general malicious software such as trojans and viruses.

Table 1: Attack Stage oriented intrusion alert classification

Alert Class	Description	Example
Recon_Sniffing	Reconnaissance step. Indicates that the attacker may be sniffing the channel. Motivation may be to tamper with network communications.	Snort alert *DNS SPOOF query response with TTL of 1 min. and no authority* with Snort ID 254 indicates a possible scanning of DNS traffic by an attacker and a possibly spoofed reply to a DNS query.
Recon_Footprinting	Reconnaissance step. Attacker gains knowledge of the target network or organization's security posture [10], e.g., identifying the organization's domain names.	Snort alert *SNMP request udp* with Snort ID 1417 indicates a possible attempt to identify which devices are using SNMP by trying an SNMP-Trap connection. This knowledge can be exploited.
Recon_Scanning	Reconnaissance step. This can happen before Recon_Footprinting as well. Generally, attacker tries to refine and verify the knowledge gained during the Footprinting phase [10]. E.g., Ping attacks. Sometimes reveals specific software details like versions. Can be used along with Footprinting to constitute a *Fingerprinting* attack. Non-intrusive.	Snort alert *ICMP PING BeOS4.x* with Snort ID 370 indicates an ICMP echo request coming from a host running BeOS4.x.
Recon_Enumeration	Enumeration [10] is another Reconnaissance step. Usually employed after previous steps. Attacker tries to identify user accounts to exploit, poorly protected resources etc. Different from previous stages as it involves active connections to targets. Information gathering stage.	Snort alert *WEB-MISC .htgroup access* with Snort ID 1374 indicates an attempt to gain group access permissions on a webserver. This gives an attacker useful information that he can use to further attack the webserver.
Intrusion_Root	Intrusive step into a target machine with the privileges of administrator. Attacker may have access to a command shell with the same privileges. May overlap with buffer overflow attacks classified here as *Escalation*, but includes attacks that may exploit configuration flaws.	Snort alert *EXPLOIT LPD dvips remote command execution attempt* with Snort ID 1821 indicates the possible exploitation of a configuration vulnerability in dvips on some Red Hat systems allowing an attacker to execute commands with administrative rights.
Intrusion_User	Intrusion with privileges of non administrative user.	Snort alert *RSERVICES rlogin login failure* with Snort ID 611 indicates that an attacker may have tried to use *rlogin* for remote login by guessing the password. Successful attack yields user permissions.
Escalation_OS	Privilege escalation step (usually buffer overflow attacks). Escalation exploits vulnerability in a specific operating system or in software usually bundled with a certain OS.	Snort alert *NETBIOS DCERPC Messenger Service buffer overflow attempt* with Snort ID 2257 indicates that an attempt has been made to exploit a buffer overflow vulnerability in Windows Messenger Service in Microsoft Windows NT and 2000.
Escalation_Service	Privilege escalation step exploiting a vulnerability (usually a buffer overflow vulnerability) in a specific service or software package, rather than a specific OS vulnerability.	Snort alert *WEB-CGI ezman.cgi access* with Snort ID 2206 indicates an attempt to exploit a buffer overflow vulnerability in EasyBoard 2000 1.27 .
Goal_DoS	Alerts that indicate the possibility of Denial of Service attacks.	Snort alert *DOS Jolt attack* with Snort ID 268 indicates that the attacker is trying to send large fragmented IP packets to the internal network, indicating a Jolt Denial of Service attack.
Goal_Ethical	Attacker's goal is purely *Ethical*. Indicated by observing that an attacker penetrates a system to the point where he can carry out malicious attacks at will, yet refrains from doing so.	Alerts that are raised, for example, during penetration testing. Such an alert may be generated by a higher-level alert fusion system rather than by a low-level sensor.
Goal_Corruption	Attacker tries to corrupt a target machine, its configuration, and/or data. Clearly a hostile action and indicates an active goal-oriented adversary with malicious intent.	Snort alert *WEBPHP phpBB privmsg.php access* with Snort ID 2078 indicates that an attacker can use a specially crafted SQL query to delete all private messages for users on the system. This represents tampering with data on the target network, and is thus classified as a corruption step.
Goal_Espionage	Goal of the adversary is *Espionage*. This involves steps like trying to obtain password files, access keys and so on.	Snort alert *FTP authorized_keys* with Snort ID 1927 indicates that an attacker may be trying to obtain sensitive information like the users and hosts allowed to connect via *ssh* to a certain machine.
Goal_Backdoor	Attempt to install a backdoor on the target machine to facilitate future attacks.	Snort alert *BACKDOOR netbus active* with Snort ID 109 indicates that the Netbus backdoor or trojan horse may be installed.
Goal_Pilfering	Attacker's goal is to pilfer, steal, and/or exfiltrate data from the target machine.	Snort alert *MYSQL root login attempt* with Snort ID 1775 indicates that the attacker may be trying to pilfer data from a MySql database.

EXPERIMENTAL EVALUATION

OVERVIEW OF FUSION FRAMEWORK

Here we present an overview of the alert fusion framework into which we deploy the alert classification system presented here. A detailed description is available in [2]. We define hierarchical templates for multi-stage goal-oriented attacks called *Scenario Graphs*. IDS alerts form the atomic level (*Attribute Node*), exploits comprising of alerts form intermediate levels (*Attack Node*), and entire attack scenarios consisting of exploits form the top-level of these templates (*Scenario Graph*). Alerts arriving in an alert stream are matched with elements of these templates to provide inferences of attack scenario development. Figure 1 depicts the hierarchical structure of this model.

Attribute Node: An Attribute Node represents an event (usually an IDS alert) in our framework. This event is represented as a collection of *Attribute Fields*, which are elements (tags) in the newly emerging Intrusion Detection Message Exchange Format (IDMEF [12]) format. Some of the Attribute Fields are designated as *Critical Fields*. An alert in the alert stream triggers an Attribute Node in a template if its Critical Field values match those defined for the latter. An Attribute Node that is triggered has its *Credibility Value* changed dynamically from 0 to 1 and contributes an *a priori* determined weight to its parent Attack Node(s).

Attack Node: An Attack Node is an exploit that is represented in terms of its components (child nodes), which may be Attribute Nodes or other Attack Nodes. An Attack Node has a correlation function that defines how the weights of its child nodes (which are triggered at a point in time) are correlated to calculate its own *Attack Node Credibility Value*. Attack Nodes are members of higher level Scenario Graphs and contribute their credibility values to them in a dynamic process.

Scenario Graph: This is a template of a multistage attack scenario which has several attack stages. These stages may typically be exploits. We view each such stage as a node (Attack Node) of the Scenario Graph. The Scenario Graphs are goal oriented, with at least one node of a Scenario Graph being identified as a goal node. A Scenario Graph also has a *Scenario Credibility Value* (zero, initially), that varies dynamically as fusion of the live intrusion alert stream proceeds. A Scenario Graph developed based on this framework is thus a complete representation of an attack scenario with its constituent exploits and intrusion alerts that indicate these exploits and incorporates the various relationships that exist between these elements.

Thus in real-time, one obtains dynamically varying credibilities of goal-oriented attacks based on the state of a live intrusion alert stream. This provides the necessary Situation Awareness to the security analyst.

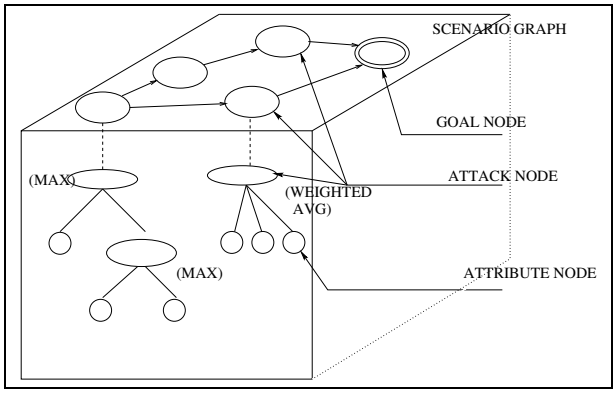

Figure 1: Hierarchical model of Fusion Framework showing a Scenario Graph and levels of Attack Nodes

EXPERIMENTS

We performed some experiments using the fusion framework and alert clustering scheme presented earlier. The network under consideration was complex and consisted of multiple subnets and hundreds of machines designed to simulate an actual military network. The experimental data consisted both of unlabeled data corresponding to multistage attacks against high-value targets in the network, and *ground truth* data which was used to verify if the detection process was successful or not.

The steps taken were as follows:

- Attribute templates (Alert data, primarily Snort alerts, numbering in the thousands) corresponding to the attacks were passed through the fusion engine.
- The fusion engine clustered alerts based on their stages in a multistage attack (i.e., based on our

scheme) and triggered Scenario Graphs with these stages as Attack Nodes.

- The resulting Scenario Graphs were analyzed by an analyst to determine what high-level attack information was provided by them.

- The results were compared with the *ground truth* to verify the correctness of the results. Analysis of whether major attack stages were detected and victim machines identified was done to determine if the system provided effective and useful Situation Awareness.

An example of the resulting Scenario Graphs (top-level template) is shown in Figure 2.

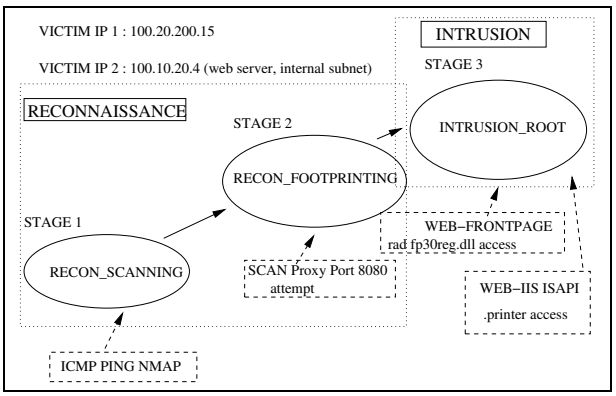

Figure 2: Effective Attack Awareness indicating attack stage progression. Same attack stages are detected against both victims.

ANALYSIS

Analysis of the results and the ground truth for the above example indicated that the multistage attack was mainly aimed at two target or victim machines: A web server in an internal subnet with IP address 100.10.20.4 and a host in a separate enclave with address 100.20.200.15. The victim machines were correctly identified, and the attack steps were analyzed to be:

- Scanning of various machines in the network (Recon_Scanning - ICMP PING)

- Identification of specific versions of services running on the target machines (Recon_Footprinting)

- Various kinds of web-attacks against the target aimed at intrusion, most of which involved a possibility of root privileges. Examples of alerts detected in this category are *WEB-FRONTPAGE rad fp30reg.dll access* (result: possible unauthorized administrative access), *WEB-IIS .printer access* (result: serious, unauthorized administrative access), *WEB-FRONTPAGE /_vti_bin/access* (result: unauthorized administrative access) and so on.

We see that the thousands of alerts pertaining to several machines are clustered using our approach into high-level attack stages providing useful information and Situation Awareness to the analyst. The analyst is informed that attackers have carried out reconnaissance (both preliminary and specific) and that there is a serious possibility of intrusion into the specified hosts with administrative access. The analyst is thus not burdened with the volume of alerts and differences in alert types (the different attacks aimed at intrusive root access indicate the same thing from a network defense point of view). Details of the alerts in each group are easily available to the analyst by drilling down the specific nodes. This provides a quick 'indications mechanism' so that the analyst can issue the necessary warnings to pre-empt attacks and protect critical resources.

RELATIONSHIP TO INDICATIONS AND WARNING FRAMEWORK

The alert clustering technique we have presented here can be integrated into the Situation Awareness and Indications & Warning Framework as advocated by the military ([7], [8]). The clustering of alerts into attack stages provides network-independent Indications of various attacks. This, when combined with network-dependent information like IP addresses and connectivity and the fusion model and templates described before, provides effective Situation Awareness to the analyst enabling him to issue an accurate Warning (E.g., a Warning could be the result of a Scenario Graph credibility value going above some threshold). This is shown in Figure 3.

CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a scheme for the classification of intrusion detection sensor alerts

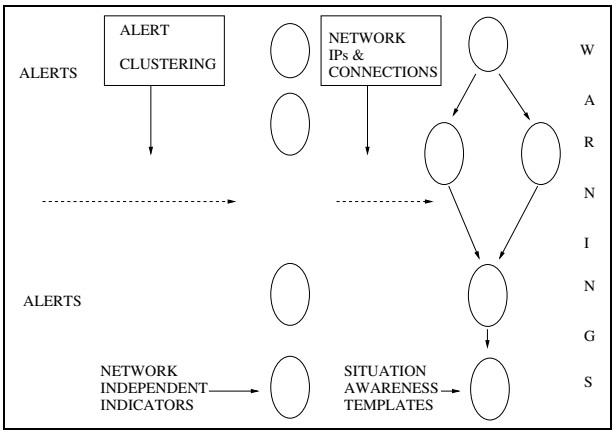

Figure 3: Relationship to Situation Awareness and I&W Framework

based on their role as part of goal-oriented multistage attacks. We have also shown some examples of Snort alert classifications based on this scheme and have demonstrated experimentally how this provides enhanced Situation Awareness. We have described how this fits into the Indications & Warning framework advocated by the military. Future work involves the mapping of sensor alerts from other IDS sensors into this scheme and the development of a fully functional and robust Situation Awareness tool. We also plan to carry out further experiments and present concrete metrics which would better quantify the usefulness of our approach. The incorporation of network details with minimum user configuration is seen as an important challenge.

ACKNOWLEDGEMENTS

The authors would like to thank John Salerno and Samuel Gorton for useful discussions during the course of this research.

REFERENCES

[1] Stefan Axelsson. Intrusion Detection Systems: A Taxonomy and Survey. *Technical Report No 99-15, Dept. of Computer Engineering, Chalmers University of Technology, Sweden*, March 2000.

[2] Sunu Mathew, Chintan Shah and Shambhu Upadhyaya. An Alert Fusion Framework for Situation Awareness of Coordinated Multistage Attacks. In *Proceedings of the 3rd IEEE International Information Assurance Workshop (IWIA 2005), College Park, Maryland*, March 2005, Pages 95-104.

[3] John D. Howard and Thomas A. Longstaff. A Common Language for Computer Security Incidents. *Technical Report SAND98-8667, Sandia National Laboratories*, October 1998.

[4] Jeffrey Undercoffer and John Pinkston. Modeling Computer Attacks: A Target-Centric Ontology for Intrusion Detection. In *2002 CADIP Research Symposium*, October 2002.

[5] P.G. Neumann and D. B. Parker. A Summary of Computer Misuse Techniques. In *Proceedings of the 12th National Computer Security Conference, Baltimore, Maryland*, October 1989, Pages 396-407.

[6] U. Lindqvist and E. Jonsson. How to Systematically Classify Computer Security Incidents. In *Proceedings of the IEEE Symposium on Security and Privacy, Oakland, CA*, May 1997, Pages 154 - 163.

[7] John J. Salerno, Michael Hinman, Douglas Boulware. Building a Framework for Situation Awareness. In *Proceedings of the International Society on Information Fusion (ISIF) Conference, Stockholm, Sweden*, 2004.

[8] John J. Salerno, George Tadda, Michael Hinman and Samuel Gorton. Achieving Situation Awareness in a Cyber Environment. *To appear at SIMA 2005, MILCOM 2005, Atlantic City, NJ*, October 2005.

[9] Edward G. Amoroso. Fundamentals of Computer Security Technology. Prentice-Hall PTR, 1994.

[10] Joel Scambray, Stuart McClure, George Kurtz. Hacking Exposed: Network Security Secrets and Solutions. Osborne/McGraw-Hill, 2001.

[11] SNORT http://sourceforge.net/projects/snort-idmef

[12] The Intrusion Detection Message Exchange Format. http://www.ietf.org/internet-drafts/draft-ietf-idwg-idmef-xml-14.txt