A GENERIC TAXONOMY FOR INTRUSION SPECIFICATION AND RESPONSE

S.M.Furnell, G.B.Magklaras, M.Papadaki and P.S.Dowland

Network Research Group, Department of Communication and Electronic Engineering, University of Plymouth, Plymouth, United Kingdom
e-mail: nrg@jack.see.plym.ac.uk URL: http://ted.see.plym.ac.uk/nrg

KEYWORDS

Intrusion Detection Systems, Intrusion taxonomy, Intrusion specification, Automated response.

ABSTRACT

The paper presents a preliminary description of an intrusion taxonomy to aid the development of a generic intrusion specification and response platform. Existing intrusion taxonomies are assessed in order to derive a suitable classification of incidents that would be both detectable and addressable by an automated intrusion detection system. The issue of automated responses to intrusions is considered, along with the factors that would influence the level of response selected. This work represents a contribution to ongoing research in relation to the Intrusion Monitoring System, a conceptual architecture for Intrusion Detection.

INTRODUCTION

For the last twenty years, the computer security world has witnessed the growth and continuous development of Intrusion Detection Systems (IDS). These tools monitor the events occurring in a computer system or network and search for indications of security-related problems. There are many challenges in the development process of these systems and, to date, the majority of research has centred around the issue of how an intrusion may be detected (Mukherjee et al. 1994). One issue that has not been conclusively addressed is the classification of different intrusions into a consistent framework that can be used as a basis for further work. With an appropriate taxonomy as the core, it becomes possible to pursue related work in relation to both the specification of, and response to, intrusions.

It is considered that a suitable specification of an intrusion (in terms of the detectable indicators) may be used as input to an IDS to enable the identification of the associated attack. At present, there is only one widely recognized theoretical study of intrusion specification, described by Feirtag et al (2000). However, the derived 'Common Intrusion Specification Language' has a number of disadvantages that might limit its application to large commercial systems. It is outside the scope of this paper to systematically discuss these disadvantages but the reader can find additional reference in (Doyle, 1999). The existence of these limitations indicates strongly the need for a more systematic examination of the foundations of an Intrusion Specification Language. It is also important for recognised intrusions to be linked to appropriate responses.

The issue of *automated* response is important for the following reasons:

- there is an increasing need to ease the load on system administrators/security architects as corporate IT infrastructures become larger and more complicated.
- many intrusion incidents are generated by automated scripts. As a result, the speed with which a response should be initiated is great. Moreover, the increase in network bandwidth coupled with the distributed nature of many attacks and the exponential growth in CPU power, narrows the margins left for a non-automated system response.

Despite this, the issue of automated response has been widely neglected in the process of developing research prototypes and commercial IDS products, the focus having been given to detecting the intrusions themselves.

This paper aims to establish the foundations for developing a generic Intrusion Specification Language and response platform at a preliminary level. The discussion begins with an outline of the Intrusion Monitoring System (IMS), a conceptual architecture that represents the focus of the research to be presented. This is followed by a brief review of existing intrusion taxonomies, leading into an overview description of a derived approach, which is considered to represent a suitable basis for considering the issues of intrusion specification and response. The issue of automated response is then considered, presenting the top-level considerations for an intrusion response framework and an example of how this could be applied in practice. The paper concludes with a look ahead to intended further research in this area.

THE INTRUSION MONITORING SYSTEM

IMS is a conceptual architecture for intrusion monitoring and activity supervision, based around the concept of a centralised host handling the monitoring of a number of networked client systems. Intrusion detection in the system is based upon the comparison of current user activity against both historical profiles of 'normal' behaviour for legitimate users and intrusion specifications of recognised attack patterns. The architecture is comprised of a number of functional modules, addressing data collection and response on the client side and data analysis and recording at the host. The roles of these modules are summarised below.

The **Anomaly Detector** analyses the data gathered by the IMS clients for signs of suspected intrusion. This data can be compared against both the user's behaviour profile and the generic intrusion specifications (i.e. attack signatures).

The **Profile Refiner** allows the automatic modification of a user's profile in response to a valid session profile. This recognises the fact that a user's behaviour pattern may change over time.

The **Recorder** stores a temporary record of system and user activity during a session (session profile) which can be used by the Profile Refiner to update the user profile, providing the session was not considered anomalous.

The **Archiver** provides an audit log, storing all security relevant events.

The **Collector** provides an interface between the IMS client and the applications running on the client computer. The collector is responsible for gathering information relevant to the user and system activities.

The **Responder** provides the interface between the IMS software suite and the end-user. Its main task is that of monitoring the signals sent from the server to the client and taking appropriate action where necessary. This will be considered further in the sections that follow.

The **Communicator** provides the interface between the client and server IMS software. The communicator is responsible for ensuring a consistent, reliable and secure exchange of data between the client and server.

The **Controller** provides a management interface, allowing an administrator to configure the IMS system-operating parameters.

The architecture is described in more detail by Furnell and Dowland (2000). For the purposes of the discussion in this paper, the key elements are the anomaly detector (which would make use of appropriate intrusion specifications derived from the taxonomy) and the responder (which deals with suspected problems).

EXISTING INTRUSION TAXONOMIES

The first step towards establishing an Intrusion Specification Language (ISL) is to derive a taxonomy of intrusive activities. A number of intrusion taxonomies have been devised to date. However, before these are considered, it is useful to define the terms 'intrusion' and 'intrusion taxonomy'. Appropriate definitions are provided by Amoroso (1999), who defines the term intrusion in an IT context as *"a sequence of related actions by a malicious adversary that results in the occurrence of unauthorized security threats to a target computing or networking domain"*. The reader will notice that this definition emphasises the existence of a set of resources, dividing them into computers and networking (telecommunication equipment that interconnects the discrete computing units). The author proceeds further and defines the term intrusion taxonomy to be a '*structured representation of intrusion types that provides insight into their perspective relationships and differences*'. In this case, the author denotes the process of spotting common or major differences between intrusions as a measure to ease the automation of a response.

There are currently, there are three widely accepted intrusion taxonomies. A brief overview of these is given below.

– **SRI Neumann-Parker Taxonomy** (Neumann and Parker, 1989): An intrusion taxonomy based on a large number of incidents reported to the Internet risks forum. The taxonomy classifies intrusions into nine categories, according to key elements that might indicate a particular type of incident. Table 1 below summarises the overall scheme.

Table 1: SRI Neumann-Parker (NP) Taxonomy

NP 1 EXTERNAL MISUSE	Non-technical, physically separate intrusions
NP 2 HARDWARE MISUSE	Passive or active hardware security problems
NP 3 MASQUERADING	Spoofs and identity changes
NP 4 SUBSEQUENT MISUSE	Setting up intrusion via plants, bugs
NP 5 CONTROL BYPASS	Going around authorised protections/controls
NP 6 ACTIVE RESOURCE MISUSE	Unauthorised changing of resources
NP 7 PASSIVE RESOURCE MISUSE	Unauthorised reading of resources
NP 8 MISUSE VIA INACTION	Neglect or failure to protect a resource
NP 9 INDIRECT AID	Planning tools for misuse

- **Lindqvist and Jonssen's intrusion taxonomy (**Lindqvist and Jonsson, 1997): This effort could be considered as an extension of the SRI Neumann-Parker taxonomy. It further refines security incidents into intrusions, attacks and breaches. It examines these issues from a system-owner point of view, based on a number of laboratory experiments. The results of these experiments indicated a need for further subdivision of the Neumann-Parker classes 5, 6 and 7, as shown in table 2 below. Their work provides further insight into the process of spotting aspects of system elements that might indicate an intrusion.

Table 2: Lindqvist and Jonssen Extension of the SRI NP Taxonomy

Extended NP5 CONTROL BYPASS	Password attacks, spoofing privileged programs, utilising weak authentication
Extended NP6 ACTIVE RESOURCE MISUSE	Exploitation of write permissions, resource exhaustion
Extended NP7 PASSIVE RESOURCE MISUSE	Manual browsing, automated browsing

- **John Howard's security incident analysis** (Howard, 1995)**:** This is focused on the process of attack, rather than classification categories. It establishes a link through the operational sequence of *tools*, *access*, and *results* that connects the attackers to their objectives. Although Howard's work cannot be considered as a pure taxonomy, the wealth of statistical analyses and the various cases mentioned provides some of the most well-written and useful material for considering/revising new taxonomies.

A PROPOSED TAXONOMY FOR INTRUSION SPECIFICATION AND RESPONSE

Although the previously mentioned taxonomies are indeed useful for the systematic study of intrusions, none of them is tailored for the purposes of producing the structure of an Intrusion Specification Language. The classification criteria employed by these taxonomies cannot be qualified or quantified very easily by an Intrusion Detection System. The best way to overcome this problem is to devise an intrusion taxonomy scheme that is based on elements of the IT infrastructure that are being targeted. The idea is that it is easier to detect which particular element is affected by an intrusive action, rather than trying to sense the origin, entity or the motives for initialising an attack. This information is also considered sufficient to determine the main options for response. As a consequence, the following target-based intrusion classification schema has been devised, based on things that could be directly detected by an Intrusion Detection System (IDS). The level of IT component granularity increases towards the bottom layers of the suggested hierarchy, all the way down to individual self-contained components. This level of granularity is necessary for devising a comprehensive Intrusion Specification Language set. However, the language itself will not be defined in this paper and, as such, the discussion will consider only the top three layers of the suggested taxonomy.

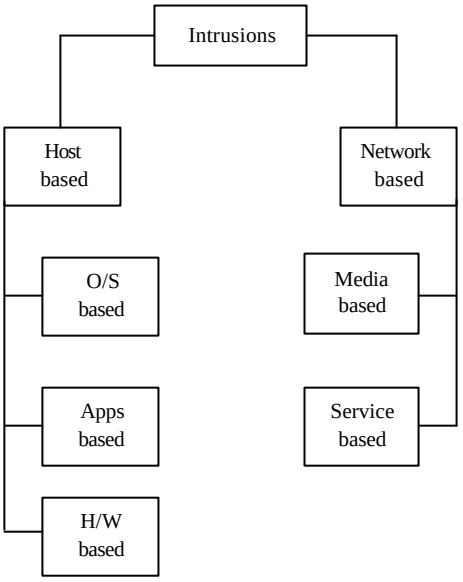

Figure 1: Levels 1 and 2 of the Taxonomy

Figure 1 indicates that, at the top level, intrusions can be sub-divided into host and network based categories. This is because certain attacks focus upon computing systems (servers, desktop workstations, thin/embedded clients), whilst there are others that target the equally important elements that interconnect them.

The host-related intrusions are divided into three major sub-categories. The operating system (O/S) based category includes all intrusive activities that aim to compromise functions such as memory management, I/O activity and file storage operations (see Figure 2). A typical example of a host-related attack could be a buffer overflow attack.

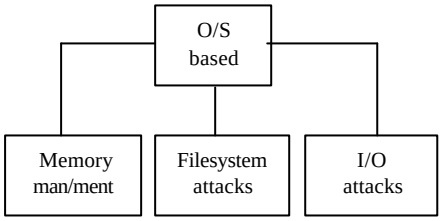

Figure 2: Operating System Intrusions

The application-based intrusion category concerns all intrusions that may affect the operation of a particular software package that is using the various operating system services, as described in Figure 2. However, this category refers specifically to files that are maintained by the application itself, rather than generic system or user data files. These files often carry a particular extension and could be manipulated in various ways in order to halt or affect the operation of the application in specific ways. For example, if a configuration file of the application is changed, then it is possible to make the application disclose confidential information. If an application log (data) file is manipulated, then valuable data might be lost or stolen (Figure 3). Although there is a substantial overlap between application and operating system intrusions, the two should not be confused. For instance, if a non-legitimate user modifies an application file, then the problem is really related to the failure of the Operating System to authenticate the file manipulation. However, if this action is initiated by a legitimate user, then the application itself should contain additional functionality to detect and contain the resulting effects and the incident should belong to the application-based category of our taxonomy.

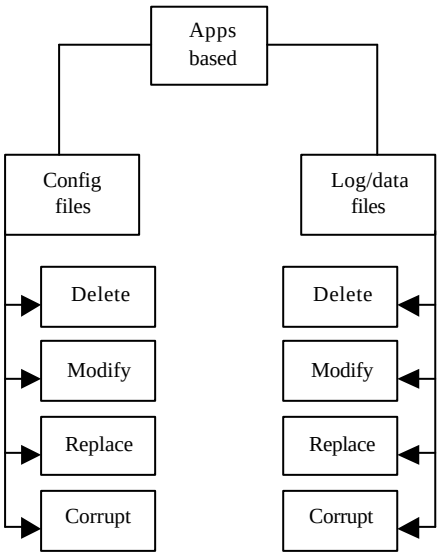

Figure 3: Application-based Intrusions

Finally, intrusive activities may concern the hardware components of a host. For instance, the non-authorised addition of a modem on a secure server may or may not provide a security threat because it opens the door to a non-secure environment such as the Public Switched Telephone Network (PSTN). Theft, vandalism and changes in the configuration of hardware components, in order to disable security features are also common scenarios, illustrated in Figure 4.

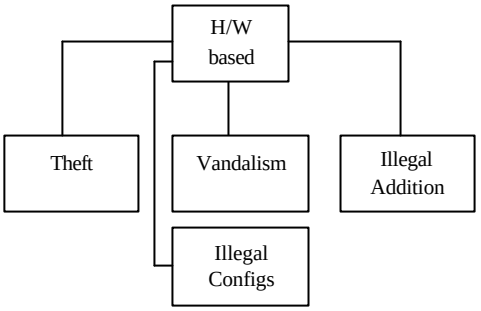

Figure 4: Hardware-based Intrusive Activities

Network-related intrusions could be further subdivided into media and serviced-based intrusions. The word 'media' encompasses all the hardware components that are responsible for the physical transfer of the network packets, whereas 'services' are discrete functions performed by specific telecommunication elements such as routers, gateways, firewalls and other devices.

In line with what can happen with host related hardware, media can be stolen, vandalised or configured in a non-authorised way. In addition, many intrusive activities tend to target the physical signaling of the medium itself, something that is not common in host-related hardware. The detection of these disruptions is still a fruitful area of research.

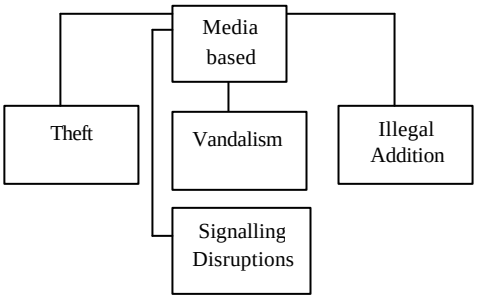

Figure 5: Network Media-based Intrusions

Finally, service-based attacks might target the smooth operation of routing and management services. The former concerns the vital operation of network equipment: without routing no network can function. The latter is also important for the smooth operation of large corporate data networks and concern tools that configure, troubleshoot and provide redundancy services (network address translation, load balancing).

As previously indicated, this classification provides a fairly high level view, but it is sufficient to begin classifying practical incidents and determine appropriate responses. For the detection of a particular intrusion, a more precise specification is necessary, requiring further levels of decomposition within the taxonomy.

AUTOMATED RESPONSES IN INTRUSION DETECTION SYSTEMS

Intrusion response can be defined as the process of counteracting the effects of an intrusion. It includes the series of actions taken by an Intrusion Detection System, which follow the detection of a security-related event. It is important to note that consideration is not only given to taking action after an intrusion has been detected, but also when events of interest take place and raise the alert level of the system. That is the early stages of an attack, when the system is suspecting the occurrence of an intrusion, but is not yet confident enough to raise an alarm.

The aims of response actions can be summarised into the following:

1. Protect system resources
   - in the short term, this will include mechanisms to contain the intrusion, as well as to recover and restore the system to a well known state
   - in the longer term, learn from the intrusion and use this knowledge to remove identified vulnerabilities of the system, and to enhance the detection and response capability. The underlying idea is to make sure that the intrusion cannot be repeated.

2. Identify the perpetrator of the intrusion.

The contribution of automated response can be mostly focused on the protection of system resources. Further investigation of the intrusion to identify the perpetrator is thought to require co-operation with other parties, like Incident Response Teams, and mostly falls under the operational aspect of response.

Issues in automated response

One of the issues we need to consider for response to intrusions is the confidence level of the system, in order to avoid false alarms. In the case of a false positive, we may find automated response itself to become a denial of service issue, by affecting the access level of legitimate users. Recommended actions to increase the certainty level are based on a combination of detection and reaction in order to collect additional information about the attack. According to the level of confidence and the seriousness of the potential intrusion, those actions could be:

- further investigate details of the intrusion in audit log files;
- record details in an intrusion log for further inspection / investigation;
- alert the system administrator and increase the intrusion alarm;
- increase the monitoring level;
- issue a challenge for further authentication;
- limit permitted user behaviour;
- delay (or lower priority of) intruder's session / process;
- terminate (or suspend) the anomalous session / process.

The severity, as well as the discrete characteristics of an intrusion, are also issues that need to be matched to the confidence level, to determine and prioritise actions of response. It is important to recognise and identify the threats posed to the system so that appropriate actions can be taken in time, to prevent the system from reaching a compromised state.

Furthermore the impact of response actions upon users and the system is another consideration that should also be taken into account. It is desirable to preserve the transparency of system response as much as possible, so that no disturbance to legitimate users will be added and no alert to attackers will be given to make them aware of the fact that they are being monitored. The latter might give attackers the opportunity to cover the traces of their activities, and possibly cause further damage to the system. Alternatively, the sooner actions are taken, the safer it is for the system to preserve its state and minimise the damage from the attack.

The overall process is illustrated in Figure 6, which indicates the inputs to an entity such as the IMS Responder and shows the possible responses that may be taken at different impact levels.

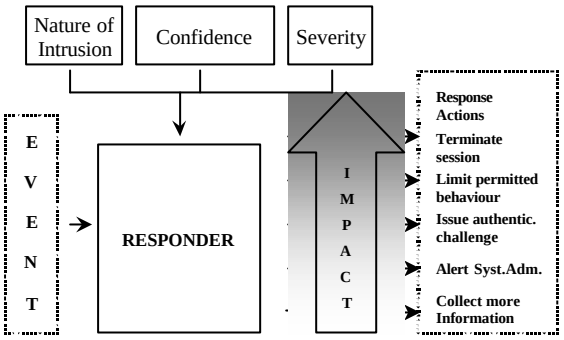

Figure 6 : Issues in Response

Example - Counteracting DoS attacks

As an example of potential response levels, this section considers the issue of Denial of Service (DoS) attacks – which would be classified as network/service-based intrusions under the earlier taxonomy. DoS attacks are an increasing threat to Internet systems, as illustrated by the fact that they account for 60% of reported incidents affecting WWW sites (Power, 2000).

Speed of detection and response is a major requirement for this class of attack. They are difficult to guard against - mainly due to the fact that they are identifiable from their results (i.e. when it is already too late to prevent them). The issue of how to respond to DoS attacks is an area of ongoing work in the research community. The most dominant approach is *resource management*, which is based on monitoring the resource requirements of computational tasks, adjusting their priorities to make sure that the capacity of the resource is not overloaded. It may include resource management for both the host and network domain, defining intra-host parameters (scheduling, storage management) and inter-host channels of allocation (task migration, network flow control) (Tung, 1997).

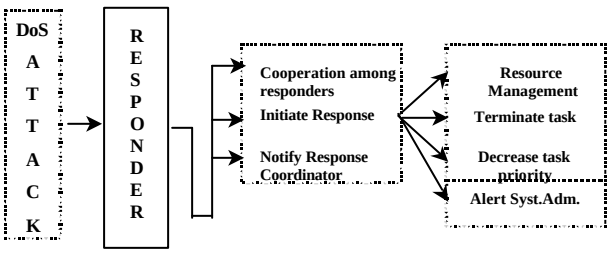

Figure 7: Response Actions for DoS Attacks

However, resource management may not be the only, or most desirable, response in any given situation. Examples of different levels of response that may be taken against a DoS attack are illustrated in Figure 7, which also indicates the stages that a Responder agent may take in a networked monitoring environment in order to mount a co-ordinated response.

CONCLUSIONS AND FUTURE WORK

The taxonomy presented in this paper provides the foundation for ongoing work in relation to the issues of intrusion specification and response.

A generic Intrusion Specification Language will be based around a full version of the taxonomy presented in this paper and will enable the description of events in a manner that is independent of particular system / application configurations. It is intended that the language will facilitate the description of both an attack and the consequent response that should be applied.

The response framework is also the focus of ongoing research. The main tasks will involve classifying the range of responses appropriate to the different categories of intrusion from the taxonomy, and then measuring the effectiveness of the different actions (considering their impact to the system/legitimate users and strength against attackers).

It is considered that cooperation between Responders residing in different networks would be a desirable feature. Coordination of those elements will then be needed and the evaluation of possible response strategies will be examined. A possible disadvantage of this approach would be the utilisation of this feature to deceive responders and utilise them either as information sources or agents to launch attacks. Thus careful consideration should be given for the secure communication between those elements.

REFERENCES

Amoroso E. 1999. 'Intrusion Detection: An Introduction to Internet Surveillance, Correlation, Traps, Trace Back, and Response' , Second Printing, Intrusion.Net Books, New Jersey, June 1999, Chapter 4, pp100-105.

Doyle, J. 1999. “Some representational limitations of the Common Intrusion Specification Language (CISL)”, http://www.medg.lcs.mit.edu/projects/maita/documents/cc2/cisl/

Feirtag R., Kahn C., Porras P., Schnackenberg D., Staniford-Chen S., Tung B. 2000. 'A Common Intrusion Specification Language’. http://www.gidos.org/

Furnell, S.M. and Dowland, P.S. 2000. "A conceptual architecture for real-time intrusion monitoring", *Information Management & Computer Security*, vol. 8, no. 2: pp65-74.

Howard, J. 1997. *An Analysis of Security Incidents on the Internet 1989 – 1995*. Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, April 1997. http://www.cert.org/research/JHThesis

Lindqvist U. and Jonsson E. 1997. "How to Systematically Classify Computer Security Intrusions", *Proceedings of the 1997 IEEE Symposium on Security and Privacy*, May 4-7, 1997, IEEE Computer Society Press.

Mukherjee, B.; Heberlein, L.T.; Levitt, K.N. 1994. "Network Intrusion Detection", *IEEE Networks 8*, no.3: 26-41.

Neumann, P.G. and Parker, D.B. 1989. "A summary of computer misuse techniques". In *Proceedings of the 12th National Computer Secuirty Conference* (Balitimore, USA, 10-13 Oct): pp396-407.

Power, R. 2000. "2000 CSI/FBI Computer Crime and Security Survey", *Computer Security Issues and Trends*, Vol. V1, No. 1. pp1-15.

Tung, B. 1997. "CRISIS: Critical Resource Allocation and Intrusion Response for Survivable Information Systems", Presentation held at *Intrusion Detection Workshop* (Savannah GA, February 1997). See http://www.isi.edu/~brian/crisis/inprint/savannah.ps

BIOGRAPHIES

GEORGE MAGKLARAS has gained a first class Honors degree in Computer Systems & Networks from the School of Computing, University of Plymouth. He has worked as a network and software development specialist at the European Headquarters of the IBM NUMA-Q team in England, where he specialised on operating system and networking support for Symmetric Multi-Processing (SMP) servers. He has also consulted on a number of UNIX/LINUX IT projects in England, Greece and the Netherlands. He has won a three-year EPSRC studentship to pursue a PhD on a 'Generic architecture for Intrusion Specification and Misuse Detection in IT systems'. His work is also supported by the Metropolitan Police Computer Crime Unit and Orange Personal Communications. He is currently working as a researcher and part time lecturer with the Network Research Group at the University of Plymouth.

MARIA PAPADAKI was born in Iraklio of Crete, Greece and studied Informatics in the Technological Educational Institute (T.E.I.) of Athens. After her graduation in November 1997, she worked for two years for the Library and the Network Operating Centre of the Athens School of Fine Arts. Funded by the State Scholarships Foundation (SSF) of Greece, she attended the MSc course *Integrated Services and Intelligent Networks Engineering* at University of Plymouth, UK (1999-2000) and is currently a research student within the Network Research Group of the University. Current interests include intrusion detection and methods of automated system response.