# MODELING AND EVALUATION OF IDS CAPABILITIES FOR PREVENTION OF POSSIBLE INFORMATION SECURITY BREACHES IN A WEB-BASED APPLICATION

**Pjotrs Dorogovs[(a)], Andrejs Romanovs[(b)]**

[(a)],[(b)]Riga Technical University, Kalku Street 1, LV-1658 Riga, Latvia

[(a)]pjotrs.dorogovs@rtu.lv, [(b)]andrejs.romanovs@rtu.lv

## ABSTRACT

Nowadays with vast growing amount of network information systems and their integration not only into work but also into people's private life assuring security of industrial and private information assets is becoming extremely sensitive and topical issue. There is huge number of available free-ware and paid methods of information protection from unauthorized access by unwanted individuals. Currently significant attention of researches in the field of information security is focused on using various intellectual data mining techniques for building an intellectual information security system. Such security systems roughly (for the purpose of this paper) can be divided into intrusion protection and intrusion detection systems – IPS and IDS.

Keywords: information security, intrusion detection, modeling of IDS capabilities, web-based application

## 1. INTRODUCTION

In general intrusion protection systems includes any available method or recommendation that prevents attackers from gaining access to secured network, system or information asset. Most common ways to ensure high availability Intrusion protection system is usage of any kind of firewall or anti-virus software. Most commonly known five types of IPSs are – inline NIDS, application-based firewalls, layer seven switches, network-based application IDSs and deceptive applications. Each type of mentioned Intrusion protection system has different level of provided protection. There is no possibility to choose best solution since all of them have their pros and cons. By performing analysis of the way each IPS works it's possible to define which exact one would fit best for your needs. Sometimes it's even advisable to build an IPS containing more than one of previously mentioned solutions. For example one of already broadly used ways is – using a layer seven switch in front of Internet firewall to defend against DoS attacks and known attacks and using application layer firewalls/IPS software or hybrid switch to protect Web servers. Currently other ways are being discovered and tested.

Intrusion detection systems, in turn, may be considered as a type of security assuring method as for information systems as also for computers. Such system should make a comprehensive analysis of gathered information of computer, network or information system activities to proactively identify potential security breaches that might include both attacks from inside and outside of protected perimeter. The underlying reason why intrusion detection systems should be used is relatively straightforward – data and systems should be protected from all information security aspects.

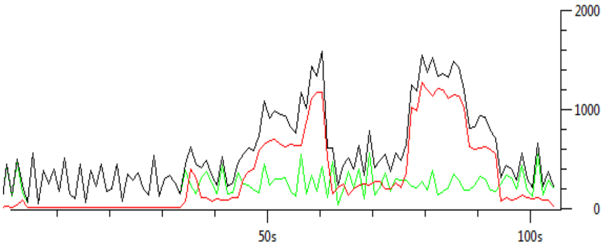

Figure 1: Network activity statistics during DoS attack

Modern Intrusion detection system must be capable of determining level of information confidentiality, integrity and availability (CIA) that are commonly referred as a kind of benchmark for evaluation of information security as such. The fact that data and systems cannot always be protected from outside intruders in modern Internet environment using ordinary security mechanisms such as password and file security, leads to a range of issues. Further measures beyond those normally expected of an intranet system should always be made on any system connected to the Internet. Intrusion detection takes that one step further. Placed between the firewall and the system being secured, a network based intrusion detection system can provide an extra layer of protection to that system. For example, monitoring access from the Internet to the sensitive data ports of the secured system can determine whether the firewall has perhaps been compromised, or whether an unknown mechanism has been used to bypass the security mechanisms of the firewall to access the network being protected. Besides that high quality IDS should assure significant level of ability to recognize user policy violations and abnormal activity patterns.

## 2. TYPES OF INTRUSION DETECTION SYSTEMS

Currently all intrusion detection systems available on the market fall into two categories – Network based systems which are placed in the network nearby system that is being monitored and that examines network traffic and Host based systems which actually run in the system being monitored and that examines activity of monitored system. Most recent type of Intrusion detection systems reside in the operating system kernel and monitor activity at the lowest available level of protected system.

Table 1: Host and network based systems

| Benefit | Host based systems | Network based systems |
|---|---|---|
| Deterrence | Strong deterrence for insiders. | Strong deterrence for outsiders. |
| Detection | Strong insider detection. Weak outsider detection. | Strong outsider detection. Weak insider detection. |
| Response | Weak real-time response. Good for long-term attacks. | Strong response against outsider attacks. |
| Damage Assessment | Excellent for determining extent of compromise. | Very weak damage assessment capabilities. |
| Attack Anticipation | Good at trending and detecting suspicious behaviour patterns. | None. |
| Prosecution Support | Strong prosecution support capabilities. | Very weak because there is no data source integrity. |

Network and host-based IDS bring very similar advantages. Both of them are very well fit for outsider deterrence. Network-based systems are able to warn attackers regarding their illegal actions thus working as a buffer for inexperienced hackers showing them that they are not as safe as it seems like. Contrarily host-based systems work on an assumption that people that are aware about constant monitoring of their actions are less likely to commit misuse. And although both type of systems are able to detect vast variety of intrusion actions first are more oriented exactly to network activities while second are able to detect more insider actions. Furthermore both systems can react and even alert security personnel about possible misuse.

### 2.1. Misuse and Anomaly detection in Intrusion detection systems

Broadly speaking modern intrusion detection systems use techniques that can be divided into two major categories: misuse detection and anomaly detection. Taking into consideration effectiveness of the anomaly detection technique not only against known types of attacks (like misuse detection does by exploiting signature database) but also against new ones, it has become a topical issue in majority of data and computer security researches.

In recent years a vast majority of research activities in the area of anomaly detection have been focused on studying the behavior of programs and the creation of their profiles based on system call log files. Until now, a simple anomaly detection method based on monitoring system calls initiated by the active and privileged processes is widely used. The profile of normal behavior is constructed by enumerating all unique, and related fixed length system calls, which are observed in the training data; in turn, previously undetermined sequences are considered abnormal. This approach has been extended by various other methods. It was suggested to utilize data mining approach to study samples of the system calls and construct small set of rules contained in normal data. During the monitoring and detection, the sequences that violate these rules are treated as anomalies. For example Hidden Markov Model (HMM) can be used – a method for modeling and evaluation of invisible events based on system calls. Later, the idea of analyzing patterns of system calls of fixed length has been further developed, by analyzing patterns of system calls, but of variable length. Furthermore, a new method for intrusion detection, based on the method of principal components has been introduced.

Profiling the behavior of the end user is not less important aspect of data protection than the profiling the software activities. This method is effective in detecting internal attacks that constitute one-third of the corporate system security. In information systems based on UNIX or Linux operating system, the sequences of shell commands are easily collectible and analyzable information, thus being the source material for creating profiles of end users. Besides, the collection of such information does not use significant system resources. On the other hand, taking into account the difference between the behaviors of end users, building of profiles of their activities is a difficult task comparing to building a profile of program behavior. Hackers can even try adapting their behavior to fool IDS systems.

## 3. IDS SOLUTIONS

Great part of modern intrusion detection methods are directed to proactive analysis of already conducted attacks and creation of new rule sets for detection of next attack of same type on the basis of previously generated rules. Scope and efficiency in such a case will be limited with those rules for specific types of attacks. Nevertheless enormous traffic caused by new attack cannot be detected. Consequently it is crucial to perform fast analysis of anomaly traffic instead of really detailed to make it possible to determine possibility of incoming anomalous traffic. Usually network traffic analysis consists of following basic functions: primitive network traffic data, integration of traffic data and detection of anomalous network behavior. The main concern with network traffic analysis is not the mere traffic counts but the definition which network should analyze traffic features so that actual collecting method and types are determined. Methods for detection of abnormal network behavior are analogous to intrusion detection system. It detects and analyses network traffic that consists of attacks based on network traffic patterns of well-known attacks. Another method for traffic classification is based on modeling of normal behavior of network activities. Both methods require modeling of network traffic and analysis of related functionality for abnormal traffic. Generally well-known tools such as Ntoop are used for traffic analysis. Besides that TcpDump, IPmon and Snort tools can be used.

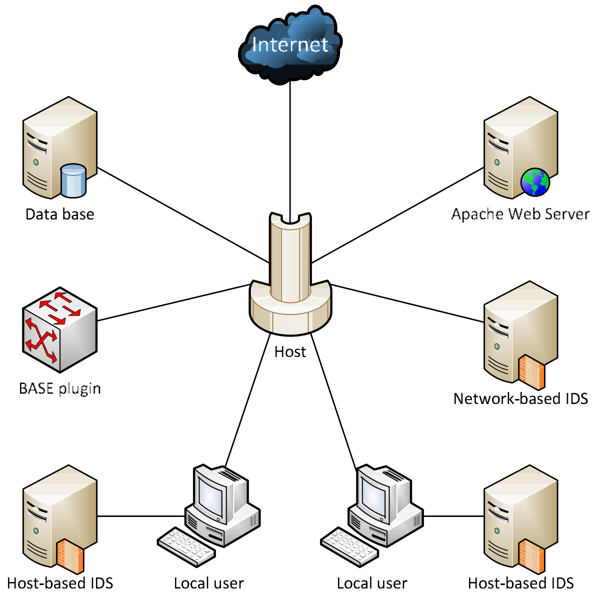

Figure 2: Snort IDS Example

Snort for example is a behavior-based and rules-based NIDS that demonstrates outstanding performance in real-time traffic analysis and packet log analysis. It can be utilized for protocol analysis, examination of packet content, pattern matching, port scan, CGI attacks, buffer overflows etc. It uses very flexible rule language consisting of a module plug-in structure to catch traffic. Three main Snort functions are:

- It can be used as a packet sniffer such as TcpDump.
- Network traffic debugging is available based on internal packet logging function.
- It shows good NIDS functionality. Snort is a packet-sniffing tool that uses the packet capture library of Libpcap.

Snort recognizes sniffed packets and compares them with pre-defined detection rules via a pre-processor and a detection engine to detect an intrusion. Rules for Snort can be easily created by users that later can be applied as a plugin operations with different alert logs and pre-processors. Nevertheless due to simple pattern-matching possibility of false-positives is quite high and detection on new attack types is almost impossible.

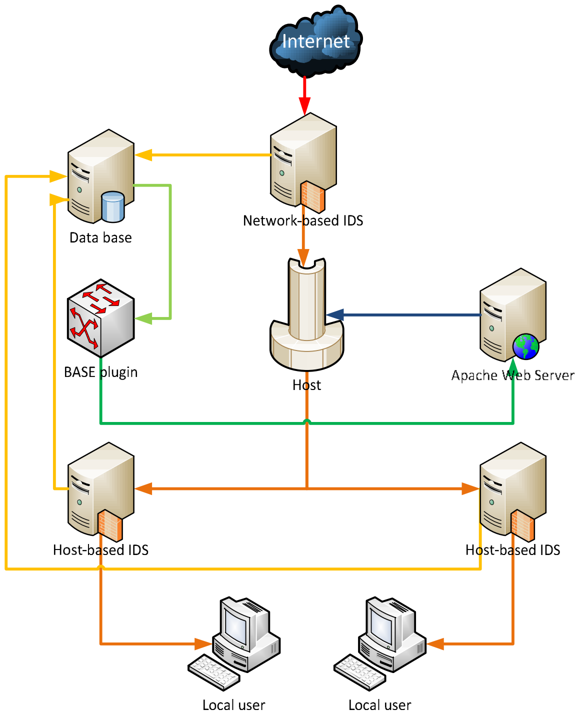

Figure 3: Snort IDS Working Principle

→ Represents incoming traffic.

→ Represents filtered traffic inside local network.

→ IDS traffic to Data base.

→ Information flow from data base.

→ Reconstruction of data to HTML/PHP.

→ Representation of filtered HTML/PHP.

Intrusion detection systems can define abnormal use, misuse, and abuse of a computer system as well as determine the proper action in the event of an intrusion. Although main purpose of intrusion detection system is detection of possible intrusion its' construction also includes active response that is based on current condition of used environment. Normally any Intrusion detection system detects an intrusion by using information from a database and notifies recognized attack attempts to an administrator.

Great part of new researches in the field of intrusion detection focuses on anomaly detection. However, a number of systems still use the detection methods against abnormal behavior similar to misuse detection.

### 3.1. Role behavior profiling

As it was mentioned earlier behavior profiling is one of the most effective methods for malware attack detection. Modern researches on the modeling of profile are mainly focused on the end-user behavior or program behavior. For dynamic environment such a web-based system adoption of end-user behavior modeling, where limited users boast the fixed activities in their daily operations is more suitable. Unfortunately for the web-based system, which is accessed by millions of users for the information every day, creation of a behavior model for each individual user is nearly impossible. Commonly, the designated CGI programs access the database as the middleware between the database and almost all users. Therefore profiling of program behavior is not significantly better either.

On the other hand three-tier architecture associated with databases mainly is set of known applications. Users interact with data base system using various operations that are either authorized by applications or by end-user itself. In any case the activity on the database strictly depends on the privilege of the executors.

In comparison to user profiling, role profiling can show more regular patterns because functions and tasks operate on interrelated data and therefore is a more static set of sequences of operations.

### 3.2. Logging policies

Enormous volumes of data in raw log files are the bottleneck place for performance and reliability of any Intrusion detection system. To overcome this problem logging policies can be introduced. One of the possible policy implementation is assigning of different levels of logging for different users. For example, Intrusion detection system will trace the most detailed information for the root user, such as remote IP address, their specific actions, etc. and in contrary least detailed information for guest user. Such technique can noticeably reduce response time of IDS, make use of log files more efficient and consequently improve overall performance of whole system.

### 3.3. Intrusion detection capabilities in encrypted web traffic

As it was mentioned previously usually an IDS is being deployed near the web server and monitors the network activities by performing protocol analysis and pattern matching. In other words, IDS should reconstruct HTTP headers and payload from captured packets, and identify attacks by comparing traffic to signatures of attacks or behavior profiles.

Such mechanisms as SSL (Secure Socket Layer) or its successor TLS (Transport Layer Security Protocol) were introduced as solution for secure communication and data transfer over the Internet. These protocols first of all allow authentication of servers and users and secondly considerably contribute for safeguarding confidentiality, integrity and availability of data. For encrypted traffic simple Intrusion detection systems need a deposited private key. Alternatively they will have to monitor traffic just after decryption.

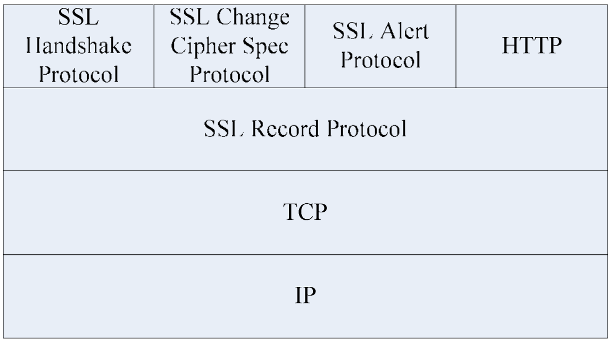

Figure 4: SSL Protocol Stack

These conventional approaches are problematic from the perspective of key management and network configuration and tuning. However such approaches are becoming more and more popular taking into account rising popularity of web systems and application that require secured communication between end-user and server. Thereby web-application server administrators are faced with the dilemma either to provide secured services using SSL/TLS protocols but with less secured system itself because of lack of IDS monitoring or vise-versa.

### 3.4. Attack types

Currently wide range of attack types against web-application is known. Below most common of them are clarified. It should be noted that such attack classification does not refer to traditional signature based approaches that compare a HTTP request strings to a set of signature strings. Also note that classification makes target clear but it is more abstract class than traditional ones. Usually systems detect such attack classes as: buffer overflow, vulnerabilities of scripting languages and scanning attacks

#### 3.4.1. Buffer overflows

Successful buffer overflow type attack allow malicious user to execute arbitrary code on the web server by overwriting stack or heap memory of the process. Though bounds of memory accesses are usually checked by running programs, unchecked memory accesses allow attackers to crash or even gain full control of a process by sending a larger request or argument. In the worst case scenario attacker can even take control of web server using such vulnerability. It is possible that web applications, modules, and script languages are also vulnerable in this way.

#### 3.4.2. Vulnerabilities of scripting languages.

Biggest part of web-applications uses such scripting languages as – PHP or Perl. Every new version of a sample of code or scripts distributed with above-mentioned languages has known vulnerabilities so they allow attacker to execute harmful codes. Hacker just has to examine which version of script is currently installed on a target server by accessing this script on this server. As soon as vulnerable script is discovered attacker can compromise server security.

Following is an example:

```
GET /adserver/adxmlrpc.php HTTP/1.0
GET /phpAdsNew/adxmlrpc.php HTTP/1.0
GET /phpadsnew/adxmlrpc.php HTTP/1.0
```

### 3.4.3. Scanning attacks

Such attacks start with examination of existence and current configuration of web or proxy server at an IP address. The attacker can obtain information about the web server and/or proxy server by using simple HTTP methods, such as GET, HEAD and OPTIONS. A directory traversal attack, which accesses the parent directory and gains information about the construction of directories and files, is also categorized as a scanning attack. The following HTTP requests are examples of scanning attacks.

```
GET http://www.smthng.com/HTTP/1.1
HEAD /HTTP/1.1
OPTIONS /HTTP/1.1
GET /HTTP/1.1
```

## ACKNOWLEDGMENTS

This research is supported by the project 2.1/ELRI - 184/2011/14 «Integrated Intelligent Platform for Monitoring the Cross-Border Natural-Technological Systems» as a part of «Estonia-Latvia-Russia cross border cooperation Programme within European Neighborhood and Partnership instrument 2007-2013».

## REFERENCES

Matthew V. Mahoney, Philip K. Chan, 2003. Learning Rules for Anomaly Detection of Hostile Network Traffic. *Third IEEE International Conference on Data Mining (ICDM'03)*

G. Vigna, W. Robertson, V. Kher and Richard A. Kemmerer, 2003. A Stateful Intrusion Detection System for World-Wide Web Servers. *Proceedings of the Annual Computer Security Applications Conference (ACSAC), pp. 34-43*

Raven Alder, etc. al., 2003. Snort 2.1 Intrusion Detection. *Syngress Publishing, Inc.*

M. Roesch. Snort - Lightweight Intrusion Detection for Networks. *Proceeding of the 13th USENIX Conference on System Admin, pp. 229-238.*

Christopher Kruegel, Giovanni Vigna, 2003. Anomaly detection for Web-based attacks. *Proceedings of the 10th ACM Conference on Computer and Communication Security (CCS'03).*

## AUTHORS BIOGRAPHY

**PJOTRS DOROGOVS** is a doctoral student at the Department of Modeling and Simulation, Riga Technical University (Latvia). He received Bachelor degree in Information Technology from Riga Technical University in 2005. He obtained Master degree in IT project management (M.sc.ing.) from Riga Technical University in 2008. His research interests are the IT security and IT governance. Since 2006 he is the Head of National Schengen Information System Unit of the Information Centre of the Ministry of the Interior of the Republic of Latvia. Since 2006 he has been participating in monthly Large-scale IT system management forum taking place mostly in Brussels organized by the European Parliament. P.Dorogovs is a member of IEEE, he participated in several international scientific conferences and research projects with scientific publications in the field of ICT.

**ANDREJS ROMANOVS** is an associate professor of Riga Technical University. Born in 1970, Riga, Latvia. Ing.oec. (1993), MBA (1995). Dr.sc.ing. (2006). He has more than 20 years professional experience in development of more than 50 management information systems for state institutions and private business in Latvia and abroad as IT project manager and system analyst. He has more than 10 years pedagogical experiences, teaching courses at the Riga Technical University. His professional interests include modeling and design of management information systems, IT governance, integrated information technologies in logistics and electronic commerce, as well as education in these areas. A.Romanovs is senior member of the IEEE, member of LSS, Council Member of RTU ITI, author of 2 textbooks and more than 30 papers in scientific journals and conference proceedings in the field of Information Technology, participated in 24 international scientific conferences, as well as in 7 national and European-level scientific technical projects.