PROCEEDINGS

55. IWK

Internationales Wissenschaftliches Kolloquium
International Scientific Colloquium

13 - 17 September 2010

Crossing Borders within the ABC

Automation,

Biomedical Engineering and

Computer Science

Faculty of
Computer Science and Automation

www.tu-ilmenau.de

TECHNISCHE UNIVERSITÄT
ILMENAU

Impressum
Published by

Publisher:	Rector of the Ilmenau University of Technology Univ.-Prof. Dr. rer. nat. habil. Dr. h. c. Prof. h. c. Peter Scharff
Editor:	Marketing Department (Phone: +49 3677 69-2520) Andrea Schneider (conferences@tu-ilmenau.de) Faculty of Computer Science and Automation (Phone: +49 3677 69-2860) Univ.-Prof. Dr.-Ing. habil. Jens Haueisen
Editorial Deadline:	20. August 2010
Implementation:	Ilmenau University of Technology Felix Böckelmann Philipp Schmidt

USB-Flash-Version.

Publishing House:	Verlag ISLE, Betriebsstätte des ISLE e.V. Werner-von-Siemens-Str. 16 98693 Ilmenau
Production:	CDA Datenträger Albrechts GmbH, 98529 Suhl/Albrechts
Order trough:	Marketing Department (+49 3677 69-2520) Andrea Schneider (conferences@tu-ilmenau.de)

ISBN: 978-3-938843-53-6 (USB-Flash Version)

Online-Version:

Publisher:	Universitätsbibliothek Ilmenau ilmedia Postfach 10 05 65 98684 Ilmenau

Home / Index:
http://www.db-thueringen.de/servlets/DocumentServlet?id=16739

DESIGN AND DEPLOYMENT OF A MONITORING SENSOR FOR ENTERPRISE'S NETWORKS

Dang Hai Hoang, Van Thuong Pham[+], Ha Duong Nguyen[++]*

* Ministry of Information and Communications of Vietnam (MIC)
[+] Vietnam Post and Telecommunications (VNPT)
[++] Faculty of Information Technology, University of Civil Engineering Hanoi

ABSTRACT

Enterprise's computers that are connected to the Internet are often targets for attacks and need to be protected. Using sensors, a network security monitoring system can collect traffic patterns from different locations in the network, analyze them and give indications of possible network attacks. However, it is still not clear how a sensor can be designed and how can the sensors be managed. This paper presents an approach for modeling a security monitoring sensor within an enterprise's network. We present an overall architecture of a sensor as well as a model for anomaly traffic behaviour detection and show possible applications to anomaly traffic detection and quality of service monitoring.

Index Terms – Network Security Monitoring, Anomaly Traffic Detection, Quality of Service Monitoring.

1. INTRODUCTION

Almost operations of enterprises are currently based on the network infrastructure. A lot of enterprise's information assets are now stored, processed and transported over networks. Attacks to these information assets are always attractive targets of hackers, either to steal information data or disturb the operations of enterprises.

The unsecured nature of network environment as well as information systems is due to the necessity for sharing and exchanging information over networks. Thus, protecting information assets becomes more critical.

Many solutions for information security have been proposed including firewalls, intrusion detection tools, malware scanning tools, etc. However, we can not only rely on these technologies due to their limitations. The major problem of these approaches is that they fail to detect new or unknown attacks.

More proactive approaches are necessary in order to be able to early detect attacks.

Network security monitoring is one of the proactive solutions. Until now there are several security monitoring frameworks that have been proposed, either based on open-source software or commercial closed software. Due to the complexity and variety of network security systems, it is difficult for enterprises to choose a suitable solution and to deploy them according to the individual needs of enterprises.

In generality, a network security monitoring system uses sensors to collect traffic data from the network, analyzes them and gives indication of possible network attacks. However, it is still not clear how a sensor can be designed and how can the sensors be deployed.

In the most cases, sensors implement intrusion detection mechanisms to detect known attacks based on signature match [1,2]. These mechanisms can not generalize the unknown attacks as indicated above.

Recently, several other works have been focused on data mining approaches [3,4,5,22]. These models try to build detection models with and without signatures. They apply data mining algorithms for large data sets that are collected from a network.

However, it is very difficult to generate required data for training process and for building pattern detection models due to the dynamic of the traffic streams. Accuracy, efficiency and usability are three remaining difficulties for the design of intrusion detection mechanisms.

In this paper, we present the overall architecture of a sensor as well as a model for anomaly traffic behaviour detection using data mining approach. We show different possible applications of the model to anomaly traffic detection and quality of service monitoring.

2. NETWORK MONITORING

Intrusion detection methods

Intrusion detection is a core problem in any network security monitoring system. Intrusion detection is aimed to collect traffic data, to analyze them to detect attacks by examining data and differentiating between normal traffic and malicious traffic (anomaly traffic).

Typical network monitoring systems have been developed until now including open source systems such as Automated Incident Reporting (Air CERT) [6], Correlated Intrusion Detection System [7], Monitoring, Intrusion Detection and Administration System [8], Sguil[9], Prelude[10], SiLK[11], OSSIM [12]…

Typically, these systems collect data from sensors, i.e. some software or hardware equipment which monitors some parts of the network. Sensors gather data, evaluate them and produce patterns for attack detection. There are usually two methods for data collecting using sensors: passive (or indirect) and active (direct). Data collection may be in real-time or using sampling method.

There are basically two principles for intrusion detection: signature-based detection (back list) and anomaly detection (white list).

Signature-based detection

This method is based on modelling known attacks. Sensors collect data from raw traffic streams being monitored and produce data patterns for detection. Detection algorithm compare sensor data to known attack patterns learned from the training data. If the collected patterns match known attacks patterns, the collected patterns can be considered as intrusion. The comparison is usually done for dependent time data series over a number of sources and protocols, in order to produce true positive decision. However, the price to paid for this method is that the data set for known attack patterns should be large enough [3]. The solution is not effective and needs much time for comparison.

Anomaly behaviour detection

Most of anomaly detection methods try to model normal behaviour of traffic (white list). Normal traffic patterns should learn from training data and the detection algorithm compare collected data from sensors to these normal patterns. If a mismatch is detected, the collected data is considered as from possible intrusion attacks. Examples of anomaly may be such as anomaly increase of traffic, or access to unallowed ports. This method is popular due to the possibility to detect unknown or new attacks [4,13,14]. However, as same as signature-based detection, it needs a large set of training data for comparison.

3. A FRAMEWORK FOR THE DESIGN OF MONITORING SENSORS

3.1. Components of a security monitoring system

Enterprise's networks are different depending on the use, necessary services, network topology, etc. However, they have common basic components such as: central system (webserver, mail server, DNS), internal links through switches and hubs, external links to Internet Services Providers.

In generality, a security monitoring system consists of two main components: data collection subsystems and monitoring center (see Figure 1).

Data Collection Subsystem

Monitoring sensors are at different locations of the network including server system, routers, switch or at endsystems. The main task of sensors is to collect data patterns or events from traffic streams and transfer them to the monitoring center.

Sensors may include traffic filters which are used to reduced the processing data. Beside of collection function, sensors can have detection function in order to reduce the collected traffic data as well as to transfer only true positive traffic data, i.e. intrusion patterns, to monitoring center (see 3.2 for details).

Monitoring Center

Monitoring center receives data patterns from sensors, stores them in a database, provides data for further analysis, e.g. correlation analyze, and performs other necessary functions such as statistical analyze, alarm generation, intrusion attack visualization, etc.

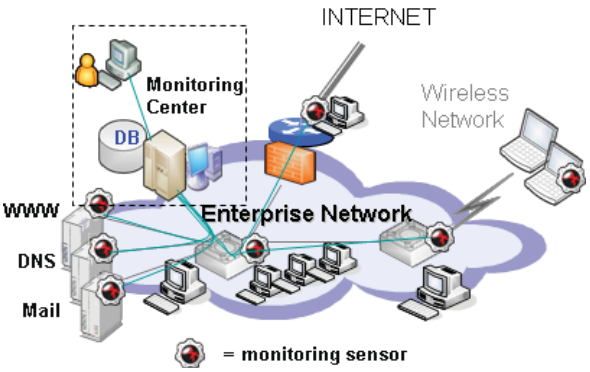

Figure 1 Security Monitoring System

3.2. Design of a Sensor

Figure 2 describes the internal design of a sensor. In this case, a sensor is a computer with the following requirements: Intel x86 compatible CPU, Processor speed at least 1 GHz, minimum 1 GBytes RAM, HDD with minimum 1 GBytes free space, two network interfaces 10/100/1000 Mbps where eth0 is used for traffic data collection and eth1 for connection to monitoring center.

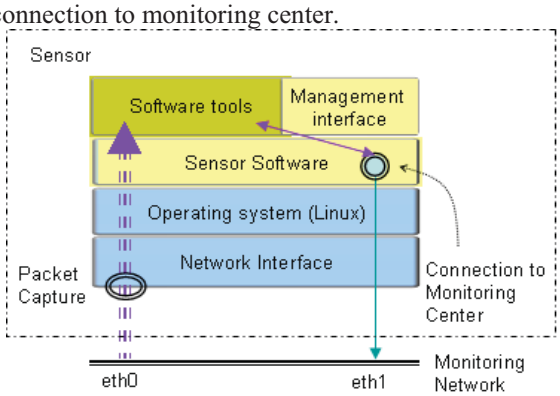

Figure 2 Internal Design of a Sensor

Sensor Software links the packet capture component to software tools, communicates with monitoring center through management interface. Sensor software can use a set of current popular open sources such as Snort, Ntop, Nagios, p0f, arpwatch, etc. [15].

Sensors have the following main functions:

+ Monitoring Traffic streams
+ Collecting traffic data

+ Detecting intrusion signatures and anomaly traffic behaviours.
+ Storing traffic patterns / intrusion patterns.
+ Managing configuration of various tools.
+ Providing interface and functions for sensor management.

4. MODEL FOR ANOMALY TRAFFIC BEHAVIOUR DETECTION

As discussed in section 1 and 2 before, intrusion detection is the key element in any security monitoring system. We also described that our sensors rely mainly on anomaly traffic detection for intrusion detection model. A question may arise how to design an effective intrusion detection model.
Most of intrusion detection models until now require data patterns for training process in order to compare the collected traffic data patterns to data patterns learned from training. Most of the models require "clean data" for training, that means data patterns of normal traffic should not include any intrusion traffic. In practice, it is difficult and expensive to produce such "clean" data patterns.
On the other hand, it should be reasonable, if data patterns for training should be produced directly from the network environment within the network to be monitored.
Therefore, a reasonable detection model should take into account the presence of noisy data, i.e. a small portion of "unclean" data together with normal traffic data. This small portion is considered as anomaly traffic. The detection model should adapt to this condition and provide usable deployment.
Several detection models based on data mining have been proposed in the past. The model in [16] makes use of log-files for anomaly detection algorithm and compares the probability of new coming traffic to probability of traffic recorded in the log-file. A mismatch indicates a possible intrusion. If no comparable traffic patterns is present in the log-file, the detection model could not work.
An adaptive intrusion detection method was proposed in [17]. This method uses influent pattern series for online detection. An expert system has been proposed to improve the method by collecting data from audit sources. Other method use continuous time series of successive packet pairs [18]. The author in [19,21] proposed to detect the frequency of occurrence of anomaly data streams in comparison to normal data streams. An expert system was proposed [20] using audit data for detection. A survey and comparison of intrusion detection methods was given in [4].
In the following paragraphs, we present an adaptive model for sensor with intrusion detection based on anomaly traffic behaviour detection in noisy traffic environment following the idea in [14] with some modifications and extension.

4.1. Components of Adaptive Detection Model

Our adaptive detection model consists of 3 modules: capture module, adaptive pattern generator, detection module (Figure 3).
The capture module collects traffic data patterns and sends them to the detection module for analyzing and intrusion pattern detecting. It also sends captured traffic data to the adaptive pattern generator for training new traffic patterns. The adaptive pattern generator is composed of: pattern receiver, pattern generator, pattern database and pattern distributor (see Figure 3).

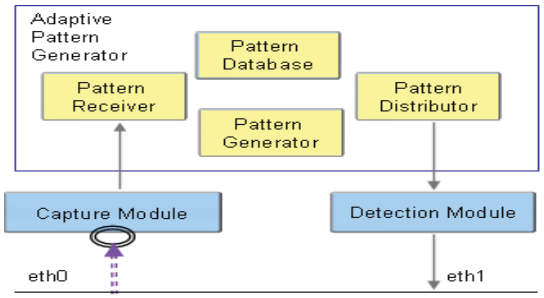

Figure 3 Adaptive Detection Model

The pattern receiver gets traffic data from capture module, i.e. from sensors on the monitoring network, converts data, stores them to the pattern database. Using pattern database queries, the set of training data pattern is produced for pattern generator. With training data from pattern database, the pattern generator builds pattern and provides them to pattern distributor, which are used by detection module. This method could be applied to any sensor device. Capture module may get raw traffic data during the time the sensor monitors traffic streams on the network.

4.2. Pattern Database

By storing collected traffic pattern from sensor in the database, we can freely access to any subset of pattern data with only unique query. Depending on various intrusion attacks, pattern database is needed to provide different pattern for training data. For instance, in case of intrusion detection based on information of system calls, we can easily model the system calls to provide suitable detection patterns. The detection algorithm can access patterns for system calls using a database query.

4.3. Pattern Generator and Pattern Distributor

We can apply any pattern generator for the model, for example the pattern database can provide training data as an input to a learning module. Whenever the pattern generator learned from training data and generates new pattern, they are sent to detection module by pattern distributor.
Two issues should be considered. Firstly, pattern generator should be able to process noisy data, i.e. data inclusive intrusion signatures. Secondly, the

detection module should be able to change detection pattern during the time. This ability determines the real-time detection possibility of the sensor.

One way to effectively implement pattern generator and pattern distributor in practice is to use distributed system. Various pattern generators and pattern distributors are distributed on different end systems. By this way, we can reduce the use of processing resources.

4.4. Anomaly Traffic Behaviour Detection under Noisy Condition

In practice, it is very difficult to guarantee "clean" traffic data by data capturing and to generate "clean" data pattern for intrusion detection. Therefore, the model should work under noisy condition, i.e. a mixture of normal data and anomaly data [14]. For a large set of traffic data, it is reasonable to consider statistical approach.

We can assume that intrusion patterns are only a small set in comparison to large set of normal traffic data. The problem becomes to detect anomaly behaviour within a traffic data set using statistical method. Pattern database may include both "clean pattern" and "unclean pattern". That is the database is a mixture of normal traffic data and intrusion data. Learning method will be applied to train the detection module over collected data in order to calculate the probability distribution of traffic data.

Under assumption of a small set of anomaly traffic behaviour data, we can assume that most of the collected data patterns are normal.

For each data pattern collected, two possibilities may happen: normal data pattern and anomaly data pattern.

Denote the probability of valid data pattern, i.e. data pattern belonging to normal traffic as P_B. Denote the probability of invalid data pattern, i.e. data pattern belonging to anomaly traffic as P_A. We have:

$P_B = (1-\alpha)$

$P_A = \alpha$

Where λ is the probability of anomaly traffic.

For any total traffic data set W, we also have a set of normal traffic data B and a set of anomaly traffic A.

$$W = P_B.B + P_A.A = (1 - \alpha).\ B + \alpha\ .\ A \quad (1)$$

The set W can be called as mixture set of the model, indicates the presence of anomaly traffic (noisy) in the set of data collected by sensors. This set depends on the probability of B and A. Each element x_i of this set W can include elements of A and B generated by the probabilities P_A and P_B.

Accordingly, we can denote B_i and A_i as the set of normal pattern elements and the set of anomaly pattern elements, respectively, at a time t, after an element x_i of set W has been processed, where *i* denotes element index and i=1 to N.

Assume that at the time of begin, no anomaly traffic occurs. The initial set of anomaly traffic A is then empty, and we have:

$$A_0 = \phi \text{ and } B_0 = W \quad (2)$$

In our model, anomaly traffic behaviour detection is equivalent to determine which elements are generated by probability distribution A and which elements are generated by probability distribution B. Elements generated by probability distribution A is considered as anomaly and elements generated by probability distribution B is considered as normal. These elements can be considered as independent statistical variables.

The detection algorithm can be formulated as follows: for each element x_i of set W (set of collected data pattern), we should determine whether it is normal or anomaly. It will be move to the set A_{t+1} if it is anomaly and moved to the set B_{t+1} if it is normal.

The probability of W at a time t is as follows:

$$P_t(W) = \prod_{i=1}^{N} P_w(x_i) \quad (3)$$

On the other hand, we have probability of B at time t:

$$P_t(B) = \prod_{i=1}^{N_1}(1-\alpha)P_B(x_i) = (1-\alpha)^{N_1}\prod_{i=1}^{N_1}P_B(x_i) \quad (4)$$

And the probability of A at time t:

$$P_t(A) = \prod_{i=1}^{N_2}\alpha P_A(x_i) = \alpha^{N_2}\prod_{i=1}^{N_2}P_A(x_i) \quad (5)$$

From (1), (3), (4) and (5) we have:

$$P_t(W) = \left((1-\alpha)^{N_1}\prod_{i=1}^{N_1}P_B(x_i)\right)\left(\alpha^{N_2}\prod_{j=1}^{N_2}P_A(x_j)\right) \quad (6)$$

Where $N = N_1+N_2$

N_1 is set of data elements generated by probability distribution of B at time *t*, N_2 is set of data elements generated by probability distribution of B at time *t*.

Equation (6) can be written in form of logarithm as follows:

$$\log P_t(W) = N_1.\log(1-\alpha) + \sum_{i=1}^{N_1}\log P_B(x_i) + N_2.\log\alpha + \sum_{j=1}^{N_2}\log P_A(x_j) \quad (7)$$

From the initial condition (2) and from (6), we get:

$$P_0(W) = (1-\alpha)^{N_1}\prod_{i=1}^{N_1}P_B(x_i) = (1-\alpha)^{N}\prod_{i=1}^{N}P_B(x_i) \quad (8)$$

By the assumption that anomaly traffic behaviours occur with a small set, i.e. α is smaller than 1-α, we can conclude: at any time t, each change in probability of the data pattern set of anomaly traffic $\left(\alpha^{N_2}\prod_{j=1}^{N_2}P_A(x_j)\right)$ has a small influence on the change of $P_t(W)$. Conversely, each change in probability of the data pattern set of normal traffic $\left((1-\alpha)^{N_1}\prod_{i=1}^{N_1}P_B(x_i)\right)$ has a large influence on the change of $P_t(W)$.

By applying the Mahalonobis distance method or the Cook distance method [23], we can suggest a

predefined parameter c and calculate the ratio of probabilities $P_t(W)/ P_{t-1}(W)$ for two successive times *t-1* and *t*.

If the ratio $P_t(W)/ P_{t-1}(W) > c$, the element x_i under consideration can be seen as anomaly traffic data element and will be moved from the set of normal data pattern B to the set of anomaly data pattern A. In the similar way, we repeat the calculation of probability ratio $P_t(W)/ P_{t-1}(W)$ for all data elements x_i in the traffic data set W.

With the assumption that the probability for receiving an anomaly traffic pattern x_i is α, the probability for receiving a normal traffic pattern x_j is 1–α, we can calculate the entropy for the set of anomaly traffic data *E(A)* and for the set of normal traffic data *E(B)* as follows:

$$E(B) = -\sum_{i=1}^{N_1}(1-\alpha)\log P_B(x_i) \quad (9)$$

$$E(A) = -\sum_{j=1}^{N_2}\alpha \log P_A(x_j) \quad (10)$$

From (7), (9), and (10), we get:

$$\log P_t(W) = N_1.\log(1-\alpha) + N_2.\log\alpha - \frac{E(B)}{(1-\alpha)} - \frac{E(A)}{\alpha} \quad (11)$$

For small set of anomaly traffic behaviour, i.e. α is more smaller than 1–α as indicated above, we can have the approximation of (11) as follows:

$$\log P_t(W) = N_1.\log(1-\alpha) - \frac{E(A)}{\alpha} \quad (12)$$

From (12) we can see that, one way to determine the probability ratio $P_t(W)/ P_{t-1}(W)$ for all data elements x_i in the traffic data set W is to determine the change of entropy *E(A)* of the data set A. If the entropy *E(A)* of the data set A at a time *t* is smaller than the one at time *t-1*, the element x_i under consideration can be seen as anomaly traffic data element.

4.5. Application of the Model to Quality of Service Monitoring

In this section, we show that the model can be easily extended for quality of service monitoring. Let we model the outlier of normal traffic as anomaly traffic, we can determine the probability of bursts of traffic data. By this way, we can determine the congestion signature as an anomaly behaviour of the normal traffic, i.e. the percentage of outlier throughput in comparison with normal throughput. If the delay of packets is considered, packets with delay larger than a threshold can be treated as anomaly traffic.

Determining the burst of traffic

Let $P_t(A)$ denotes the occurred traffic burst at time *t*, let *n* the set of traffic data belonging to the traffic burst, α is the probability that the data pattern x_i is belonging to the traffic burst, we have:

$$P_t(A) = \prod_{i=1}^{n}\alpha P_A(x_i) = \alpha^n \prod_{i=1}^{n} P_A(x_i)$$

In analogy, with $P_t(A)$ the normal traffic, m the set of normal traffic, we have:

$$P_t(B) = (1-\alpha)^m \prod_{j=1}^{m} P_B(x_i)$$

If we assume that the incoming traffic follows poisson distribution with the expected rate λ_k, *k* is the number of data patterns within the burst, we can have a simple form for anomaly traffic, i.e. the traffic burst as follows:

$$P_t(A) = \alpha^n \prod_{i=1}^{n} e^{-\lambda}\sum_{i=1}^{k}\frac{\lambda^i}{i!}$$

$$P_t(W) = \left((1-\alpha)^m \prod_{j=1}^{m} e^{-\lambda}\sum_{j=1}^{k}\frac{\lambda^i}{j!}\right)\left(\alpha^n \prod_{i=1}^{n} e^{-\lambda}\sum_{i=1}^{k}\frac{\lambda^i}{i!}\right)$$

Determining the violation of packet delay

Let $P_t(A)$ denotes the probability of delayed packets at time *t*, let *n* the set of packets experienced a delay larger than D, α is the probability that the collected packets is belonging to the this set. Assume that the incoming traffic follows exponential distribution with the expected rate λ, we have:

$$P_t(A) = \alpha^n \prod_{i=1}^{n}(1-e^{-\lambda D}) = \alpha^n (1-e^{-\lambda D})^n$$

$$P_t(W) = \left((1-\alpha)^m \prod_{j=1}^{m}(1-e^{-\lambda D})\right)\left(\alpha^n \prod_{i=1}^{n}(1-e^{-\lambda D})\right)$$

$$P_t(W) = (1-\alpha)^m (1-e^{-\lambda D})^{m+n}\alpha^n$$

For example, if α = 0.1, n = 1 (considering one individual packet), we have:

$$P_t(A) = 0.1 * (1-e^{-\lambda D})$$

$$P_t(W) = \frac{9}{10^{N+1}}(1-e^{-\lambda D})^{N+1}$$

Figure 4 and 5 show the probability distributions of $P_t(A)$ and $P_t(W)$, respectively.

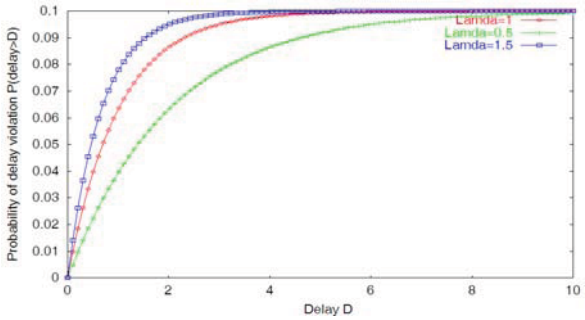

Figure 4 Probability distribution of P(A)

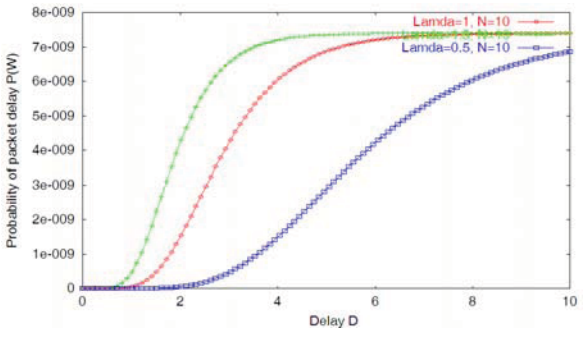

Figure 5 Probability distribution of P(W)

5. EXPERIMENT

This section presents an experiment to demonstrate the design and deployment of our sensors. In this experiment [24] we use one monitoring center with IP address 192.168.5.70. Four sensors with IP addresses 192.168.70.5, 192.168.0.120, 192.168.5.61, 192.168. 5.131. The computer 192.168.70.1 generates intrusion traffic.

Figure 6 shows the normal traffic behaviour on our experiment network in every time frame of 10 minutes. Figure 7 shows anomaly traffic behaviour generated by the intrusion computer. In the Figure 7, we can see a burst of traffic (anomaly) due to the severe requests of the remote attack computer.

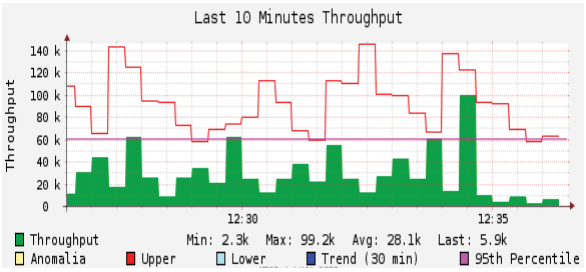

Figure 6 Normal traffic load

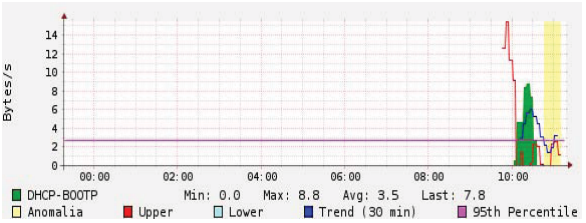

Figure 7 Anomaly traffic detection

6. CONCLUSION

The paper discussed the problem of design and deployment of a security monitoring sensor. We presented an overall architecture of a sensor as well as an adaptive model for anomaly traffic behaviour detection using data mining approach. The paper showed analytical results for the model and demonstrated different possible applications of the outlier detection model to anomaly traffic detection and quality of service monitoring. Further works can focus on investigating the performance of the outlier detection model and its applications.

7. REFERENCES

[1] Chr. Fry, M. Nystrom, "Security Monitoring", O'Reilly Media Inc. Feb.2009.

[2] R. Bejtlich, "The Tao of Network Security Monitoring Beyond Intrusion Detection", Addison Wesley, Jul.2004.

[3] W. Lee, S. J. Stolfo, and K. Mok, "Data mining in work flow environments: Experiences in intrusion detection", In Proceedings of the 1999 Conference on Knowledge Discovery and Data Mining (KDD-99), 1999.

[4] Christina Warrender, Stephanie Forrest, and Barak Pearlmutter, "Detecting intrusions using system calls: alternative data models", In Proceedings of the 1999 IEEE Symposium on Security and Privacy, pages 133–145. IEEE Computer Society, 1999.

[5] E. Eskin, M. Miller, Z.-D. Zhong, G. Yi, W.-A. Lee, and S. Stolfo, "Adaptive model generation for intrusion detection", In Proceedings of the ACMCCS Workshop on Intrusion Detection and Prevention, Athens, Greece, 2000.

[6] "The Automated Incident Reporting project", (http://aircert.sourceforge.net/)

[7] "The Crusoe Correlated Intrusion Detection System", http://www.derkeiler.com/

[8] "The Monitoring, Intrusion Detection and Administration System", (http://midas-nms.source forge.net/)

[9] http://sguil.sourceforge.net

[10] http://www.prelude-technologies. com

[11] http://tools.netsa.cert.org/silk/

[12] http://www.alienvault.com/

[13] D.E. Denning, "An intrusion detection model", IEEE Transactions on Software Engineering, SE-13:222–232, 1987.

[14] Eleazar Eskin, "Anomaly detection over noisy data using learned probability distributions", In Proceedings of the Seventeenth International Conference on Machine Learning (ICML-2000), 2000.

[15] http://sectools.org/

[16] H. S. Javitz and A. Valdes, "The nides statistical component: Description and justification", Technical report, SRI International, 1993.

[17] H. S. Teng, K. Chen, and S. C. Lu, "Adaptive real-time anomaly detection using inductively generated sequential patterns", In Proceedings of the IEEE Symposium on Research in Security and Privacy, pages 278–284, Oakland CA, May 1990.

[18] S. A. Hofmeyr, Stephanie Forrest, and A. Somayaji, "Intrusion detect using sequences of system calls", Journal of Computer Security, 6:151–180, 1998.

[19] P. Helman and J. Bhangoo, "A statistically base system for prioritizing information exploration under uncertainty", IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 27:449–466, 1997.

[20] M. Sobirey, B. Richter, and M. Konig, „The intrusion detection system aid. architecture, and experiences in automated audit analysis", In Proc. of the IFIP TC6 / TC11 International Conference on Communications and Multimedia Security, pages 278 – 290, Essen, Germany, 1996.

[21] T. Lane and C. E. Brodley, “Temporal sequence learning and data reduction for anomaly detection”, ACM Transactions on Information and System Security, 2:295–331, 1999.
[22] W. Lee and S. J. Stolfo, “Data mining approaches for intrusion detection”, In Proceedings of the Seventh USENIX Security Symposium, 1998.
[23] K.Ho, J.R.Naugher, “Outlier Lies: An Illustrative Example of Identifying Outliers and Applying Robust Models”, Multiple Linear Regression Viewpoints, Vol.26(2), pp.1-5, 2000.
[24] H.D.Hai, H.H.Thanh, N.C.Tien, B.T.Phong, N.H.Duong, N.T.Giang, “Network Security Monitoring Solutions for Enterprises”, Journal of Information and Communications Technology, Vol.2, pp.35-41, May 2010.