A NETWORK-BASED ANOMALY DETECTION SYSTEM USING MULTIPLE NETWORK FEATURES

Yuji Waizumi, Yohei Sato and Yoshiaki Nemoto
Graduate School of Information Sciences, Tohoku University
6-6-05, Aramaki-Aza-Aoba, Aobaku, Sendai-shi, Miyagi, 980-8579 Japan

Keywords: Anomaly Detection, Multiple Network Features, Intrusion Detection System, Principal Component Analysis.

Abstract: Accuracy of anomaly-based intrusion detection greatly depends on features, the numerical values representing characteristics of network traffic. In order to increase accuracy, it is necessary to choose appropriate features that can correctly detect anomalous events. In this paper, we stress the fact that a specific kind of anomaly changes specific features. We propose a highly accurate and robust intrusion detection system using multiple features. Each feature is used for evaluating anomalous events independently by a statistical detection method. Through experiments, we investigate the accuracy of the proposed scheme.

1 INTRODUCTION

The use of Internet has expanded to global scale. Along with its growth, network crimes and illegal accesses are also on the rise. Network Intrusion Detection Systems (NIDSs) are commonly used to defend against such crimes. The two most common detection techniques adopted by NIDSs are signature based detection and anomaly based detection. Signature based detection techniques search for characteristics of known attacks. Although this technique can precisely detect illegal accesses defined in the signature database, it can not detect novel attacks.

The anomaly detection technique defines the normal state of the network traffic. It regards any network state deviating from the normal state as anomalous. Thus, anomaly detection technique can be used to detect unknown attacks. However, this method has also high detection error rate because it is difficult to precisely define the normal network state. Although many researches have been carried out in the anomaly detection field (Debra et al., 1995) (SPADE,) (Mahoney and Chan, 2001) (M.Mahoney, 2003) to reduce the detection error, the detection accuracy is insufficient.

A more detailed traffic information is necessary to build an advanced anomaly detection system. In this paper, we propose a new anomaly detection technique based on multiple features of network traffic. The proposed features are selected based on three different type of attack characteristics. The three features are 1) the number of packets, 2) characteristics of packets of flow units and 3) histogram of character code of payloads. To detect any anomalous traffic, we adopt Principal Component Analysis.

2 THREE DIFFERENT FEATURES BASED ON CHARACTERISTICS OF ATTACKS

The attacks of (DARPA, 1999) are classified into four types, Denial of Service (DoS),Probe,Remote to Local (R2L),User to Root (U2R).

In order for NIDS to detect attacks, the traffic should be studied in a different basis. We reclassify these attacks based on the type of anomalies they create, as follows:

C1 Anomaly in the amount of traffic and the range of communication (DoS, Probe)

C2 Anomaly in communication procedures (Probe)

C3 Anomaly in content of communication (DoS,R2L,U2R)

Waizumi Y., Sato Y. and Nemoto Y. (2007).
A NETWORK-BASED ANOMALY DETECTION SYSTEM USING MULTIPLE NETWORK FEATURES.
In *Proceedings of the Third International Conference on Web Information Systems and Technologies - Internet Technology*, pages 410-413
DOI: 10.5220/0001279304100413

We next propose three feature sets corresponding to the above anomalies.

2.1 Timeslot Type Feature Set

Based on C1, we define Timeslot type feature set which numerically expresses the amount of traffic and the range of the communication. This feature set expresses the state of the network as a 34-dimensional vector. The Timeslot type feature set is extracted by counting the following items at fixed interval of time T_t. # of TCP, UDP, ICMP packets (3 elements), # of bytes sent and received through all TCP connections (1 element), # of TCP ports, # of occurrences of TCP flags (5 elements), # of DNS (UDP, port 53) packets (1 element), # of fragmented packets (1 element) and # of values of the 4 fields of IP addresses.

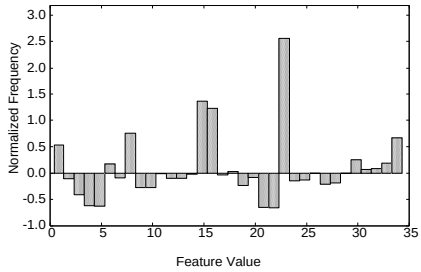

Figure 1: Timeslot type feature set of normal network state.

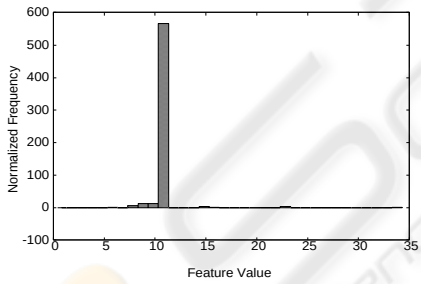

Figure 2: Timeslot type feature set of scanning ftp services.

The Timeslot type feature set is suitable for the detection of attacks which bring about changes in the amount of traffic. Examples of such attacks are scanning specific port numbers and Flood DoS. Figure 1 and 2 show a normal network state and a probe traffic, respectively. From these figures, we can see that a few specific elements extracted attack traffic(figure 2) are extremely larger than other elements compared to the normal network state(figure 1).

2.2 Flow Count Type Feature Set

Flow Count type feature set is defined from C2 of the reclassification to express the state of a flow by calculating the number of packets, flags, etc. Here, a flow is defined as the aggregation of packets which have same attributes defined by using 5-tuple (protocol,source and destination IP addresses, source and destination port numbers) (Brownlee, 1998). The start and the end of a TCP Flow are decided depending on flag bits. The end of a UDP flow, thus, is decided by setting timeout T_u.

The Flow Count type feature sets of TCP and UDP are 19-dimensional vector and 7-dimensional vector, respectively, and are defined as follows. Items about a TCP flow are # of packets (1 element), # of flows of the current flow's port number (1 element), # of fragmented packets (1 element), # of occurrences of the 8 TCP flags (8 elements) and # of occurrences of packets with only one kind of flag (8 elements). Items about a UDP flow are # of packets (1 element), # of flows of the current flow's port number (1 element), # of fragmented packets (1 element), # of sent and received packets (2 elements) and # of sent and received bytes (2 elements).

This feature set is defined to detect attacks which bring about anomalous changes in the flow structure. In other words, this feature set is defined to detect attacks which contain anomalous sequence of the flags and access to the ports which are not used by normal programs. Figure 3 and 4 depict examples of Flow Count type feature sets. Some specific elements of the port sweep traffic are extremely high.

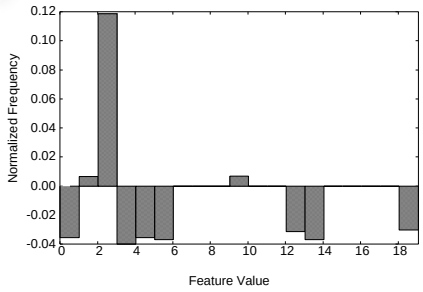

Figure 3: Flow Count type feature set of normal flow.

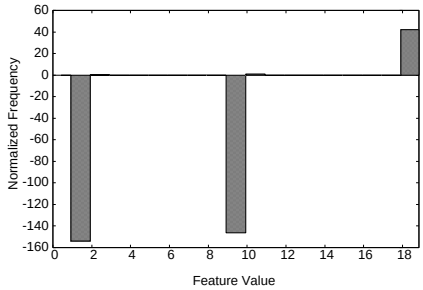

Figure 4: Flow Count type feature set of port sweep.

2.3 Flow Payload Type Feature Set

From reclassification C3, we define Flow Payload type feature set to detect anomaly of the transmitted data by calculating the character code distribution of their payloads. This feature set consists of the ratio of appearance frequencies of 8-bit codes of the flow payload. This feature is extracted from the traffics from client to server and from server to client separately. Hence this feature is expressed by a 512-dimensional vector.

The items of the Flow Payload type are as follows:

- The appearance probability of each code in the traffic from client to server (256 elements)
- The appearance probability of each code in the traffic from server to client (256 elements)

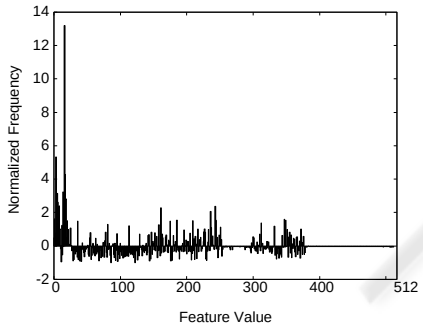

Figure 5: Flow Payload type feature set of a normal flow.

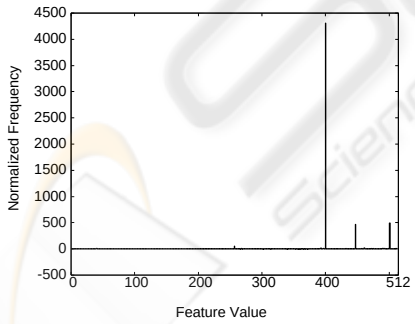

Figure 6: Flow Payload type feature set of a flow of imap attack.

Figure 5 and 6 show a normal flown and a flow of imap attack, respectively. Some elements of Flow Payload type feature set of an imap attack are exceptionally high. It is considered that these high values indicate buffer overflow traffic. Examples of such attacks are worms and DoS attacks that exploit vulnerabilities of softwares and insert large amounts of codes in order to provoke buffer overflow.

3 DETECTION OF ATTACKS BY USING MULTIPLE MODULES

The proposed method detect attacks by combining the results of modules which are defined per each feature set, as explained in **2**. We unify the results of these modules and obtain the final result. While detecting the attacks by using Flow Payload type feature sets, we use the five subsets of TCP flows which are related to the ports 20, 21, 23, 25, 80. Consequently, the number of modules corresponding to each feature set of the proposed method are as follows.

- Timeslot type - 1 module
- Flow Count type- 2 modules
- Flow Payload type - 6 modules

The final result of the proposed detection system is the `logical OR` of the detection results of these modules. That is, if any one of these modules identifies an anomaly, the system generates an alert.

The detection modules of the proposed system adopt the detection method described in (OIKAWA et al., 2002) which uses Principal Component Analysis (PCA). This method undergoes through Learning phase and Detecting phase. At Learning phase, the principal component axis is obtained from learning data by using PCA. This axis shows the characteristics of variance involving the correlations of normal traffic. At detection phase , distances from the principal component axis (Projection Distance) are calculated as *Anomaly Score* for newly extracted feature sets. When the anomaly score of a feature set surpasses a threshold, it is judged as attack at Detecting phase.

4 VERIFICATION AND EVALUATION OF THE PROPOSED SYSTEM

4.1 Experimental Environment

In this experiment, we use the data set in (DARPA, 1999) which includes five-week data. We carry out experiments taking two scenarios in mind. In scenario 1, we use the data of week 1 and 3 (attack-free) together for learning, and the data of week 4 and 5 (including attacks) for detection test. In scenario 2, to confirm our system's ability to detect attacks even if attack-included data is used for learning, data of each day of week 4 and 5 are used for learning as well as for detecting attacks. Of course, the information of

Table 1: Detection Results (Scenario 1).

Method	Detection rate(%)
Proposed Method	60.8%(104/171)
(M.Mahoney, 2003)	71.4%(132/185)
(Tyson et al., 2000)	55.6%(15/27)
(Neumann and Porras, 1999)	50.3%(85 / 169)
(Vigna et al., 2000)	46.8%(81 / 173)
(Barbara et al., 2001)	40.2%(41 / 102)

Table 2: Detection Results (Scenario 2).

Method	Detection rate(%)
The proposed System	58.5%(100/171)
NETAD	37.8%(70/185)

the attacks are not used for learning. In both scenarios, we normalize all elements of feature sets to zero mean and unit variance.

The parameters setup of the experiment are follows: # of permissible false alarms is 10/day(R. and et al, 2000), time-slot interval is 60 seconds, time out (T_u) of UDP flow is 600 seconds and validity time (T_f) is 600 seconds.

The total number of false alarms permitted in two weeks(10 days) is 100 (10 per day). The threshold of projection distance for each day for each module is determined by preliminary experiment.

4.2 Detection Performance

The detection results of the proposed system and conventional systems are shown in Table 1. The proposed system has better results for both the number of attacks detected and the detection rate compared to other methods, except for NETAD (M.Mahoney, 2003). The total number of each method is different because each method observes different objects.

The detection results of the proposed system and NETAD for scenario 2 are shown in Table 2. Table 2 shows that the detection number and the detection rate of NETAD, which shows best performance when attack-free data is used for learning, have greatly decreased. On the other hand, the detection result of the proposed system hardly deteriorates and the detection number and detection rate are higher than those of NETAD. Most of the anomaly-based IDS require attack-free data for learning. But in practice, such attack-free data are very hard to get and it is thought that the learning data that such IDSs learn have at least a few attacks. The results of scenario 2 are close to those of a real network. Therefore, the detection ability of the proposed system does not deteriorate much even under an environment close to that of a real network, scenario 2. This proves that our proposed system has high accuracy and robustness.

5 CONCLUSION

In this paper, we have proposed a anomaly detection system using three different feature sets which are extracted based on the reclassification of attacks. Our proposed method effectively detects wide range of attacks by independently treating the feature sets, and suppresses the negative effect of attack traffic included in learning data by using a statistical method in learning phase.

We have demonstrated that our proposed system can achieve high detection rate and high robustness by experiments using the data set in (DARPA, 1999).

REFERENCES

Barbara, D., Jajodia, S., Wu, N., and Speegle, B. (2001). Adam: Detecting intrusions by data mining.

Brownlee, N. (1998). Network management and realtime traffic flow measurement. *Journal of Network and Systems Management*, 6(2):223–227.

DARPA (1999). Mit lincoln laboratory - darpa intrusion detection evaluation. `http://www.ll.mit.edu/IST/ideval/`.

Debra, A., F.Lunt, T., Tamaru, H. J. A., and Valdes, A. (1995). Detecting unusual program behavior using the statistical component of the nextgeneration intrusion detection expert system(nides). Technical report.

Mahoney, M. V. and Chan, P. K. (2001). Detecting novel attacks by identifying anomalousnetwork packet headers. Technical report.

M.Mahoney (2003). Network traffic anomaly detection based on packet bytes. In *ACM-SAC*, pages 346–350.

Neumann, P. and Porras, P. (1999). Experience with emerald to date. In *Proceedings of First USENIX Workshop on Intrusion Detection and Network Monitoring*, pages 73–80.

OIKAWA, T., WAIZUMI, Y., OHTA, K., KATO, N., and NEMOTO, Y. (2002). Network anomaly detection using statistical clustering method. Technical report.

R., L. and et al (2000). The 1999 darpa off-line intrusion detection evaluation. 34:579–595.

SPADE. `http://www.silicondefense.com/software/spice/`.

Tyson, M., Berry, P., Williams, N., Moran, D., and Blei, D. (2000). Derbi: Diagnosis, explanation and recovery from computer break-ins. Technical report.

Vigna, G., Eckmann, S., and Kemmerer, R. (2000). The stat tool suite. In *Proceedings of the 2000 DARPA Information Survivability Conference and Exposition (DISCEX)*.