PCAV: Internet Attack Visualization on Parallel Coordinates*

Hyunsang Choi and Heejo Lee **

Korea University, Seoul 136-713, South Korea
{realchs, heejo}@korea.ac.kr

Abstract. This paper presents PCAV (Parallel Coordinates Attack Visualizer), a real-time visualization system for detecting large-scale Internet attacks including Internet worms, DDoS attacks and network scanning activities. PCAV displays network traffic on the plane of parallel coordinates using the source IP address, destination IP address, destination port and the average packet length in a flow. These four values are used to draw each flow as a connected line on the plane and surprisingly a group of lines forms a particular shape in case of attack. Thus, a simple but novel way of displaying traffic reveals ongoing attacks. From the fact that numerous types of attacks form a specific pattern of graphs, we have developed nine signatures and their detection mechanism using an efficient hashing algorithm. Using the graphical signatures, PCAV can quickly detect new attacks and enables network administrators to instantly recognize and respond to the attacks. Another strength of PCAV comes from handling flows instead of packets. Per-flow visualization greatly reduces the processing time and further provides compatibility with legacy routers which export flow information such as NetFlow in Cisco routers. We have demonstrated the effectiveness of PCAV using real attack traffics.

1 Introduction

Explosive expansion of computer networks has the benefit of providing much improved accessibility to a wide array of valuable data. However, the number of incidents is increasing over time. Many intrusion detection technologies have been proposed, but still have some inherent weaknesses. Conventional intrusion detection systems, which are based on known attack signatures, cannot detect unknown attacks. Further, intrusion detection systems based on anomaly detection mechanisms often generate a huge number of false alarms which overwhelm security engineers. Moreover, conventional monitoring systems such as IDS's and firewalls, provide a rudimentary level of displaying results visually.

One promising approach is visualization to handle complex situations, using a simple and intuitive method [4]. Several approaches to information visualization are studied widely [2], due to the well-known advantages resulting from visualization. Visual

* This work was supported in part by the ITRC program of the Korea Ministry of Information & Communications.

** To whom all correspondence should be addressed.

S. Qing et al. (Eds.): ICICS 2005, LNCS 3783, pp. 454–466, 2005.

images can be obtained from raw data using computer graphics techniques and algorithms. From these images, valuable insights can be acquired. It is the efficient link from the human mind to the modern computer, which represents key technology for extracting information. This visual representation is becoming more and more essential in the field of network intrusion detection [10].

We introduce a simple but novel way for visualizing Internet attacks on parallel coordinates. Parallel coordinates have many good properties such as representing more than three fields in a two dimensional space [1]. In order to visualize most notorious attacks such as Internet worms, we have carefully selected four important fields available on each flow. As a result, we can develop nine graphical signatures to detect ongoing attacks, which include Internet worms, DDoS attacks and network scanning attacks. As well, we devise an O(1) hashing algorithm to identify these signatures. The effectiveness of the proposed visualization approach is shown by running on real network traffic and revealing hidden attacks as a visual way.

We use flows for input data, instead of packets, because of system performance and compatibility with legacy routers. A flow is a single network connection and can consist of millions of packets. Handling flow-level information greatly reduces the processing time so that enables to run on high-speed links. Furthermore, many legacy routers provide flow information and they are widely deployed, which includes Net-Flow in Cisco routers. This compatibility with legacy routers greatly enhances the usability of the visualization mechanism.

The aim of this study is not to propose a new visualization technique. The main contribution is how to use parallel coordinates to display Internet attacks. Displaying network flows using carefully chosen values forms a unique graphical image for each attack. It is shown that this mechanism works for detecting notorious Internet attacks such as rapidly spreading Internet worms.

The following section shows the benefits of visualization approaches. The characteristics of Internet attacks are discussed, in particular, what data should be visualized and how this data should be visualized to make a unique shape. Section 3 describes some patterns of visualized graphs and signatures from the patterns. Then, we discuss several data structures and propose a hash algorithm to identify attack signatures. The evaluation of the mechanism is done in Section 4.

2 Attack Visualization

Humans, as a visual being, can easily recognize and infer patterns from visual aids intuitively. This section describes the benefits of attack visualization, and the principles of our visualization approaches used to display Internet attacks.

2.1 Benefits of Attack Visualization

There are four main benefits when applying information visualization to the problem of intrusion detection. First, attack visualization can easily deal with highly heterogeneous and noisy data. Network traffic is extremely complex and must be correlated with several variables such as source address, destination address, port number, packet length,

and TCP flag, but attack visualization enables us to present the traffic situation in an intuitive way. Second, attack visualization requires no understanding of complex mathematical or statistical algorithms. Visual images give perceptual clues faced with an attack. Third, attack visualization allows us to gain valuable insight into the analyzed data and deduce new hypotheses. Therefore, even though an unknown attack may have occurred, if an image pattern (signature) from the unknown attack is obtained, the attack can be quickly detected. Finally, attack visualization can be much faster than other anomaly detection approaches. Many anomaly detections require training and comparing with history, but attack visualization can quickly identify an attack using pre-defined image patterns.

2.2 Attack Characteristics

In order to devise a visual mechanism for most popular Internet attacks such as DDoS attacks, worm attacks, or network scans, their characteristics must be considered in terms of visualization. Fortunately, these notorious attacks have one common characteristic, which is called ”one-to-many relationship” between attackers and victims. While legitimate flows have one-to-one relationship, attack flows have a one-to-many relationship. This is a good point for visualizing the attacks.

A DoS attack is an attack on a computer system or network that causes a loss of service to users, typically the loss of network connectivity and services by consuming the bandwidth of the targeted network or overloading the computational resources of the targeted system. In a distributed DoS attack, the attacking hosts are often personal computers with broadband connections to the Internet that have been compromised by viruses or Trojan horse programs, allowing the perpetrator to remotely control the machine and direct the attack, often through a botnet. With enough slave hosts, the services of even the largest and most well connected website can be denied. Therefore, in a DDoS attack, there are many attackers and one victim, which forms a one-to-many relationship between a victim (destination) and the attackers (source).

A worm is defined as self-propagating malicious code. Once a machine is infected, target hosts are acquired by pseudo random number generators, or host scans to detect vulnerable machines (usually with a single vulnerability) in a certain network. If targets are decided at a previous step, an infected machine propagates worm code to targets. Therefore, a worm represents a one-to-many relationship between the infected machine (source) and the victims (destination).

Network scanning, used by hackers to probe hosts, also exhibits steps in worm propagation, and has a one-to-many relationship between a hacker (source) and scanned network hosts (destination). Port scanning is a method used for probing available services of a certain host. There is one attacker, one or many hosts, and many scanned ports. Therefore, a port scan is also characterized as a one-to-many relationship.

2.3 Four Fields as Attack Parameters

In the previous subsection, an important characteristic was presented, namely, one-to-many relationship. Now, we should consider which manifested variables display these important characteristics of attacks.

First, an attack may consist of attacker(s) and victim(s), therefore, we select the source IP address and destination IP address in a flow information. These two values can be used to visualize the particular characteristic of attacks. These values are also stored in the fields of every packet header so that they can be used to distinguish the attacking packets from legitimate packets.

Second, an attack usually targets one or more ports in TCP or UDP protocols so that the destination port number is selected as a parameter. This value identifies the targeted service of an attack and verifies port scanning attacks. On the contrary, the source port number is less meaningful than the destination port number since the source port is chosen randomly among available ports.

Third, the average size of packets in a flow can be used as a parameter. This information gives some clues whether the flow is suspicious or not. Network scanning and DDoS attacks exploit a flooding procedure and the procedure normally uses empty packets without payload. These attacking packets usually have a packet length of 40 or 48 bytes. Contrarily, Internet worms have a payload to exploit the vulnerabilities they can use. These worms are characterized by a fixed length of code because the worm codes on the payload are unique for each worm. Polymorphic worms may change their code, as an attempt to evade signature-based systems, but until now, every active worm has characterized by a unique payload.

Finally, we picked the TCP flag as an additional parameter. Network scanning and DDoS attacks may send one packet repeatedly so that its TCP flag has the same value. Thus, we can classify attack traffic or normal traffic using its TCP flag. For instance, some normal traffic have a one-to-many relationship, such as P2P communications, and they can be considered as legitimate traffic after comparing their TCP flag with that of a normal TCP handshaked flow.

2.4 Per-Flow Visualization

A visualization system can use a flow, instead of a packet, as a basic unit of data because it drastically reduces processing time without loss of necessary information. Furthermore, per-flow visualization provides the compatibility with legacy routers so that we can deploy the system without the change of current networks. One good example is to run a visualization system with Cisco routers which exports NetFlow information [5]. This implies the visualization system can use Cisco routers as sensors.

A flow can be defined as a set of packets with the same source IP, destination IP, source port and destination port that can be thought of as a connection between two computers. Available information on each flow includes the above mentioned attack parameters which are source address, destination address, destination port, average packet length, and the cumulative OR of TCP flags.

2.5 Parallel Coordinates

One important aspect of information visualization is scalability. Parallel coordinates provide great scalability to multiple dimensions. Parallel coordinates are not complex and allow multi-parameter patterns to be analyzed. We use parallel coordinates and a scattered plot matrix on the coordinates as the initial plane for visualizing Internet

attacks. This combined method also results in a quicker understanding and a more informational graph over that of a scattered plot matrix.

This visualization technique has many advantages and they are listed as follows. First, this technique does not give preference to any specific dimension. An important feature, if there is no evidence regarding which dimension is more important, is that by default there is no bias towards any specific dimension. Second, both methods have no limit to the number of parameters that can be visualized. Therefore, we can deal with a number of variables that we desire to visualize and can easily add new visualization parameters. Third, both mechanisms prominently show trends, correlations and divergences from the raw data. Therefore, this advantage enables us to gain critical insight into the flow of data and present reliable intuitive hypotheses. Using this method, even if an unknown attack occurs, a specific image pattern is obtained from the unknown attack and the attack can be detected in a timely manner. Fourth, both techniques can handle even continuous and categorical data (though some of the important benefits may be lost).

3 Parallel Coordinates Attack Visualization (PCAV)

3.1 Attack Signatures

Here we show how parallel coordinates can be used to describe an attack in a more informational graph pattern. The coordinates represent four different parameters in a flow. The first represents the source address of each flow, the second represents the destination address, the third represents the destination port and the fourth represents the average packet length of each flow. These four values shown in each flow, enables the flow to be shown as a connected line with parallel coordinates. Therefore, one connected line represents one connection. As shown in Section 2.2, the attack characteristics, of each attack have a one-to-many or many-to-one relationship with each parameter.

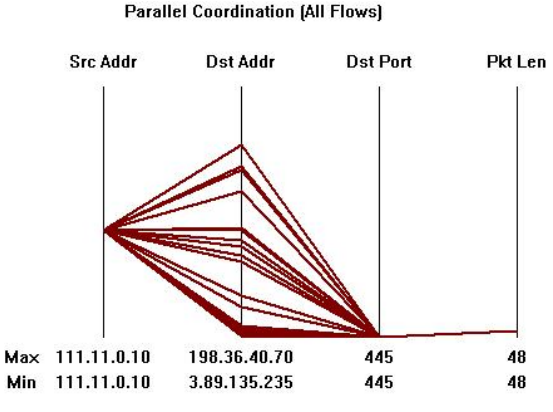

Fig. 1. The attack graph of a host scan

In order to know the graph pattern for each attack, let us consider a host scanning attack. In a host scan, an attacker may want to know which hosts are alive in the target network. The sequence of a host scan progresses gradually to check the targeted destination port of each host. Every packet usually has no payload, for more effective scans. Thus, the sizes of scanning packets can be 40 or 48 bytes only for TCP and IP headers. Occasionally scanning programs also use the TCP option of selective acknowledgment (SACK), which is commonly used to allow senders' TCP to employ more advanced loss recovery and congestion control. In these cases, the sizes of scanning packets become 48 bytes. The graph of host scans looks like a fish (diamond-line pattern) as shown in Fig. 1.

Table 1. Attack signatures of nine attacks

Implied Attack	Signature	Divergences
Portscan		1:1:m:1
Hostscan		1:m:1:1
Worm		1:m1:1
Source-spoofed DoS (port fixed)		m:1:1:1
Backscatter		1:m:m:1
Source-spoofed DoS (port varied)		m:1:m:1
Distributed hostscan		m:m:1:1
Network-directed DoS		m:m:m:1
Single-source-spoofed DoS		1:1:1:1

Not only host scans but also other attacks in one-to-many relationships show an interesting graph pattern, which are shown in Table 1. This table describes the attack signature of each implied attack and its divergences (patterns of one-to-many and many-to-one relationships). For example, in a port scan, there is one attacker and one victim, and the attacker wants to know which ports are open. To accomplish this, the attacker may use a port scanning program which checks the destination port of the victim one by one, sequentially or randomly. This behavior represents 1:1:many:1 patterns and the signature graph pattern looks similar to a kite (line-diamond) as shown in Table 1.

Average packet lengths can be used to distinguish similar attack patterns. A worm and a host scan have the same graph patterns (diamond-line). But a host scan may have no payload, whereas a worm should have a payload to infect other machines. Thus, the average packet lengths of all flows in a worm epidemic are constant and relatively larger than the average packet length of a flow in a host scan, which is 48 bytes in Fig. 1.

Backscatter is not actually an attack, but a reflective state. This state can be used to detect attacks, so it is added to the signature table. The source spoofed DoS (port fixed) is a DDoS attack using a fixed port. Usually the attack has no payload so the graph pattern is represented as a triangle with a connected line. However, a source spoofed DoS (port varied) is a DDoS attack using randomly chosen destination ports and as usual, has no payload. Therefore, the represented graph looks like a rightward looking fish. A distributed host scan is multiple host scanning behavior. Network-directed DoS is a DDoS attack targeted at a specific network, so destination addresses (victims) are network scale.

3.2 System Design

In order to display and detect ongoing attacks using the attack signatures, we propose parallel coordinates attack visualization (PCAV). PCAV has two main modules, the analyzer and the visualizer as shown in Fig. 2. The analyzer receives the flow information from the sensor, and checks whether it contains a pattern matched with an attack signature. If a set of flows form a pattern matched with an attack signature, then it implies that the attack is currently ongoing. After that, the attack data is sent to the visualizer.

The visualizer displays flow data using parallel coordinates, where flow data can be obtained from the sensor or the analyzer. Flow data including both of legitimate and attack flows comes together from the sensor. But only attack data comes from the analyzer. We can store the attack data in a database for recording the attack and it can be useful for further investigation of the incident.

The sensor can be a host or a router which generates flow-level data from network traffic. A host can run a monitor program such as nProbe [7]. A router can be enabled to generate flow data such as NetFlow information in Cisco routers.

Rescaling properties of parallel coordinates can be used to magnify an attack graph. In absolute coordinates, the top and bottom values of each coordinate are fixed in constant. Thus, it has an advantage to estimate the region of the four parameters in an attack graph. However, the unfolded portion in a coordinate is too narrow to be recognizable

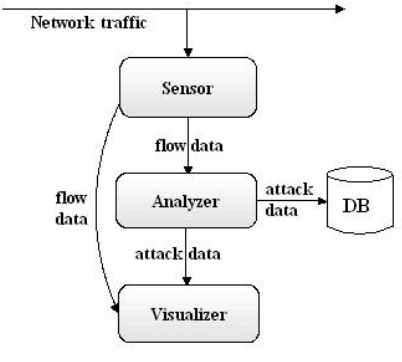

Fig. 2. System design of PCAV

by human, then, even if a detected attack was visualized, the attack graph may not show an obvious attack signature. Therefore, the visualizer in PCAV provides the rescaling operation by fitting the minimum and maximum value of each coordinates so that we can see an apparent attack signature in the attack graph.

3.3 Data Structure

In this subsection, we explain an attack detection algorithm which is running in the analyzer. The detection algorithm uses three hash tables for storing flows with respect to their source address, destination address and destination port, respectively. Then, the hash tables are used to determine which pattern in the attack signatures the flows have.

There are several data structures for storing flow information, which include linked lists, trees, and hash tables. A particular type of tree called MULTOPS [3] was also proposed for storing IP addresses efficiently. A comparison of hash tables with other structures such as linked lists, balanced binary trees, and MULTOPS trees, is shown in Table 2. From this comparison, we chose hash tables because they do not require huge memory space but provide fast lookup.

Fig. 3 shows the proposed attack detection algorithm. This algorithm consists of two parts. The first part is to generate a flow ID for an input flow using three hash tables, which is described at Step $1 \sim 12$ in Fig. 3. The second part is to handle suspicious flows using the flow ID, which is described at Step $13 \sim 30$ in Fig. 3. A flow ID has three tuples and a legitimate flow ID is either [0, 0, 0] or [1, 1, 1]. Otherwise, the flow is suspicious. If one flow comes from the sensor, the detection algorithm in the analyzer inserts the source address, destination address and destination port of the flow to each hash table. Then, the three hash tables are used to generate the three tuples of the flow ID. If an input value already exists in its hash table, then the tuple value becomes 1. Otherwise, becomes 0. For example, at time T1, if an input flow has a source IP address 1.2.3.4, destination address 5.6.7.8 and destination port 80, and at time T2, an input

Table 2. Comparisons of data structures

Algorithm	Pros	Cons	Complexity (lookup)
Linked List	Easy to implement Easy to use	High lookup complexity	$O(n^2)$
MULTOPS's tree data structure	4 lookup for searching (high speed)	1. Weak to Source spoofed DDoS attack (resource exhausted) 2. Memory usage (normal)	O(1)
Binary search tree (Balanced)	n level lookup memory usage	Lookup Complexity	O(nlogn)
Hash table	high speed	Computational overhead (function excution)	O(1)

```
Attack-Detect (F_n)   F_n ← Flow data
1    SA_n, DA_n, DP_n ← Source IP, Destination IP, Destination port of F_n
2    T_s, T_d, T_p ← Hash tables for SA_n, DA_n, DP_n
3    Attack_ID ← 0x0111  Initialize Attack_ID
4    IF hash_insert(SA_n, T_s) = TRUE ← SA_n is a new value of hash table T_s
5       Attack_ID = Attack_ID XOR 0x0100
6    ENDIF
7    IF hash_insert(DA_n, T_d) = TRUE
8       Attack_ID = Attack_ID XOR 0x0010
9    ENDIF
10   IF hash_insert(DP_n, T_p) = TRUE
11      Attack_ID = Attack_ID XOR 0x0001
12   ENDIF
13   IF Attack_ID = 0x0011
14      DDoS_fix ← F_n , DDoS_fix ← Temporary DDoS attack (fixed port) queue
15   ENDIF
16   IF Attack_ID = 0x0010
17      DDoS_var ← F_n , DDoS_var ← Temporary DDoS attack (varied port) queue
18   ENDIF
19   IF Attack_ID = 0x0101
20      IF AvgLen_n > 48  AvgLen_n ← Average packet length of F_n
21         Worm ← F_n , Worm ← Temporary Worm queue
22      ELSE Hostscan ← F_n , Hostscan ← Temporary hostscan queue
23      ENDIF
24   ENDIF
25   IF Attack_ID = 0x0110
26      Portscan ← F_n , Portscan ← Temporary portscan queue
27   ENDIF
28   IF Attack_ID = 0x0100
29      Backscatter ← F_n , Backscatter ← Temporary backscatter queue
30   ENDIF
END of Attack-Detect
```

Fig. 3. Attack detection algorithm in the analyzer

flow has a source IP address 1.2.3.4, destination address 5.5.5.5 and destination port 21, then the second flow at T2 has a flow ID of [1, 0, 0]. If at time T3, an input flow has source IP address 3.4.5.6, destination address 5.6.7.8 and destination port 80, then its flow ID becomes [0, 1, 1].

There are 6 prepared queues for each detectable attack. Once the input flow is classified into a suspicious flow in the first phase, then the suspicious flow is inserted into an attack queue corresponding to the flow ID. For instance, if a flow ID is [0, 1, 1], then we insert it to the DDoS (port fixed) queue. If the size of an attack queue exceeds its threshold for a given period, then it is considered as the occurrence of the attack. Fig. 4 shows whole stages of the attack detection algorithm in the analyzer.

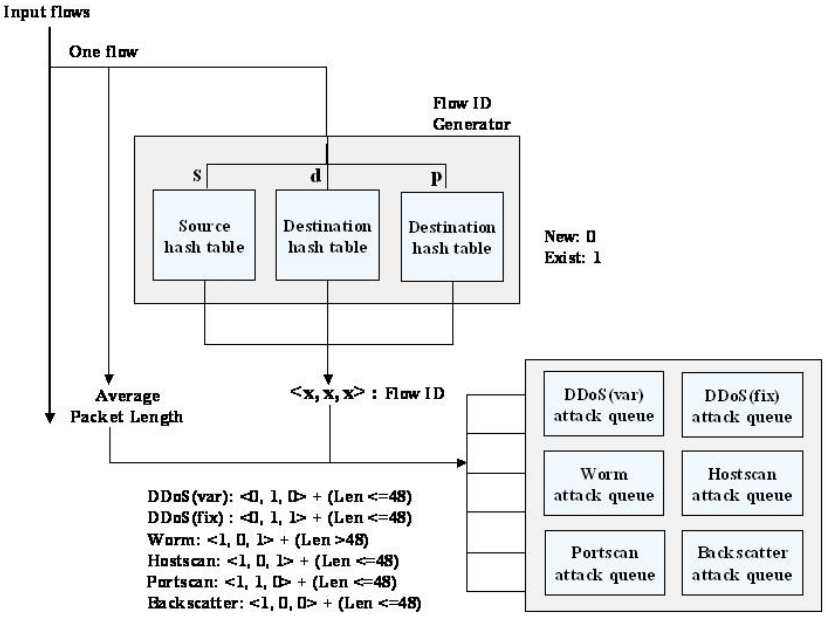

Fig. 4. Algorithm of PCAV

4 Evaluation

4.1 Attack Graphs

In order to evaluate the effectiveness of PCAV, we have implemented the PCAV system on Microsoft Windows. Attack situations are generated by replaying recorded real attack traffic. The attack traffic includes DoS attacks, SQL Slammer worms and network scanning traffic. DDoS attacks and SQL Slammer worms were captured during the incidents at one company network, and network scanning traffic was captured by using a public scanning tool. The PCAV system detects these attacks effectively and displays proper attack graphs as shown in Fig. 5.

All attack graphs are well-matched with the signature patterns shown in Table 1. The first graph is the DoS attack generated by a Blaster worm. Notice that the traffic is not a worm traffic but DoS traffic which generated by worm. The DoS attack uses a fixed destination port so the pattern represents a triangle with a connected line. The second graph is the attack graph by a Slammer worm, representing a noticeable pattern. The Slammer worm attempts to infect other machines chosen randomly so that the destination addresses should be in a random distribution. However, the pattern of the represented graph looks like a subnet scanning in the range of multicast IP addresses. This is due to a bug at the part of random number generation in the Slammer code, which generates only multicast address ranges in a certain condition. Even in this unusual situation, PCAV also detects the worm on the limited range of destination addresses.

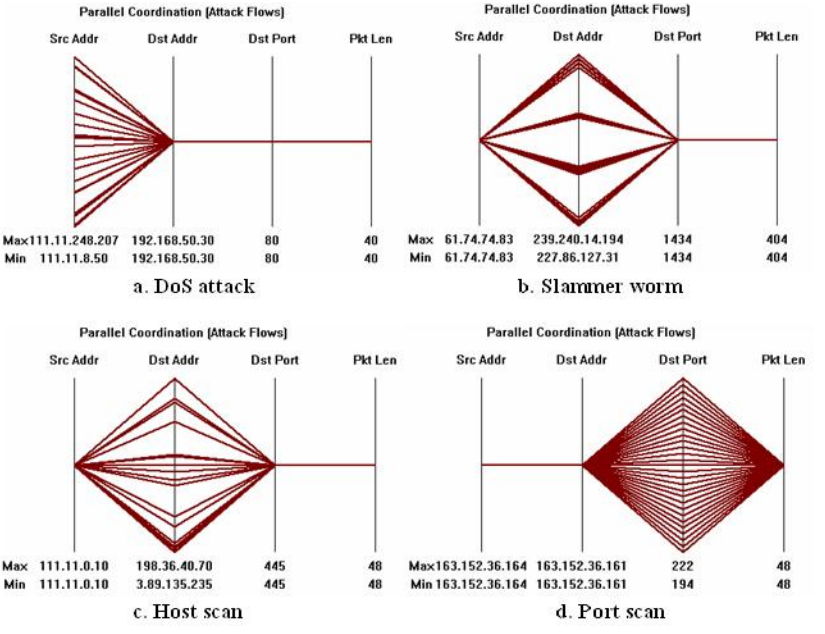

Fig. 5. Rescaled attack graphs

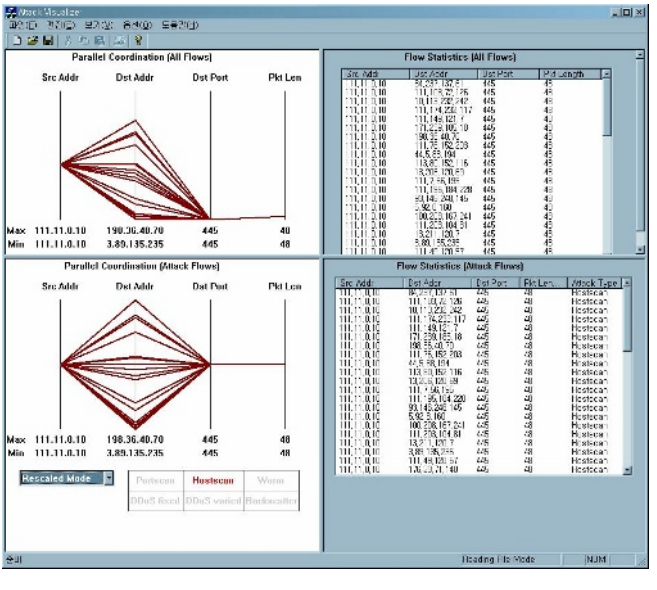

Fig. 6. Screenshot of PCAV 1.0

The third graph is generated by the PCAV system and it is the rescaled attack graph of the host scan in Fig. 1. The fourth graph is an attack graph of a port scan.

PCAV is implemented as a Windows application. PCAV can get flow data locally or import NetFlow data from remote routers. PCAV version 1.0 is shown in Fig. 6.

4.2 False Alarm Reduction

The rate of false alarm is an important metric used to measure the performance of an intrusion detection system. PCAV is sensitive to the threshold, which is closely related to the rate of false alarms. If a threshold is too high, then false positive will decrease but false negative will increase, and visa versa. Thus, we need to determine a proper threshold with the consideration of the size and the bandwidth of a monitoring network. Larger networks need to have the higher threshold.

We can further reduce false alarms by using additional parameters such as the cumulative OR of TCP flags. As well, more parameters can be added to parallel coordinates if they can enhance the correctness of attack detection.

5 Related Work

There are a number of popular monitoring tools such as FlowScan [8] and AutoFocus [9], used as traffic analyzers. Flowscan is a open source software that analyzes Net-Flow data and provides visualization graphs over five-minute intervals. AutoFocus automatically clusters traffic flows and infers patterns from the traffic.

In addition, there are several academic and commercial attempts to bridge information visualization to the field of intrusion detection. However, few of them provide real-time visualization and analyzing functionalities. One related work, regarded as previous work of this research, is 3-D visualization using a source address, destination address, destination port for detecting scanning and DDoS attacks [10]. But they cannot detect Internet worms properly and unable to distinguish legitimate traffic, such as P2P communications, from attacks.

Ourmon, which is developed at Penn State University, is an anomaly detection system using TCP flags and traffic volume as visualization sources. Ourmon presents simple bar graphs, however PCAV provides graphs which can be much more intuitively understood. Mazu Network's ProfilerTM [11], a commercial product, uses graphical profiling ability for network traffic in terms of IP address, protocols, ports, and flow volume.

Parallel coordinates are also used in other studies such as SHADOW [12], which was created at the Dahlgren Division of the Naval Surface Warfare Center. Packet headers meeting certain pre-defined boolean rules are dumped to a web-based file for examination by a human operator. SHADOW provides two visualization methods of colored histograms, parallel coordinates, and clustering methods are also used to solve various intrusion detection problems.

6 Conclusion

PCAV is a real-time visualization system for anomaly detection of Internet attacks. PCAV visualizes Internet attacks using four header fields (source IP address, destina-

tion IP address, destination port, packet length) from a flow and displays on parallel coordinates. Attack graphs generated from PCAV have specific patterns because the visual nature of the generated graphs is specific to each attack. PCAV enables the network administrator to rapidly detect and respond to malicious attacks. Even though an unknown attack may occur, specific characteristics are represented visually, and are detectable by PCAV. A plan to adopt pattern recognition methods in computer graphics areas, recognizing signatures generated by PCAV, is currently being designed. Also, visualization research regarding Spam mail distributions, P2P traffics, botnets and new types of DDoS and worm attack, is currently being undertaken.

References

1. A. Inselberg: The plane with parallel coordinates. The Visual Computer 1(1985) 69–91
2. Information visualization resources, http://www.infovis.org
3. T. Gil, M. Poletto.: MULTOPS: a data-structure for bandwidth attack detection. USENIX Security Symposium (2001)
4. D. Keim.: Visual exploration of large databases. Communications of the ACM (2001) 38–44
5. Cisco NetFlow, http://www.cisco.com/warp/public/732/Tech/netflow
6. S. Axelsson.: Visualization for intrusion detection: Hooking the worm. ESORICS (2003)
7. nProbe, http://www.ntop.org/nProbe.html
8. D. Plonka.: Flowscan: A Network Traffic Flow Reporting and Visualization Tool. USENIX LISA (2000)
9. C.Estan, S.Savage and G.Varghese.: Automatically Inferring Patterns of Resource Consumption in Network Traffic. ACM SIGCOMM (2003)
10. H. Kim, I. Kang, and S. Bahk.: Real-time Visualization of Network Attacks on High-speed Link. IEEE Network Magazine (2004)
11. Mazu Network Profiler, http://www.mazunetwork.com
12. J. L. Solka, D. L. Marchette, and B. Wallet.: Statistical visualization methods for intrusion detection. Computing Science and Statistics (2000)