This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2008 proceedings.

A Principal Components Analysis-based Robust DDoS Defense System

Huizhong Sun
ECE. Polytechnic University
Brooklyn, NY 11201 USA
hzsun@antioch.poly.edu

Yan Zhaung
ECE. Polytechnic University
Brooklyn, NY 11201 USA
yzhuan01@students.poly.edu

H. Jonathan Chao
ECE. Polytechnic University
Brooklyn, NY 11201 USA
chao@poly.edu

***Abstract*—One of the major threats to cyber security is the Distributed Denial-of-Service (DDoS) attack. In our previous projects, PacketScore, ALPi, and other statistical filtering-based approaches defend DDoS attacks via fine-grain comparisons between the measured current traffic profile and the victim's nominal profile. These schemes can tackle virtually all kinds of DDoS attacks, even never-before-seen attack types, due to the underlying statistics-based adaptive differentiation. The viability of those aforementioned statistical filtering defense systems is based on the premise that attackers do not know the victim's nominal traffic profile and, thus, cannot fake legitimate traffic. However, a sophisticated DDoS attacker might circumvent the defense system by discovering the statistical filtering rules and then controlling zombies to generate flooding traffic according to these discovered rules. This type of sophisticated attack seriously threatens the current Internet and has not yet been solved.**

In this paper, we propose a Principal Components Analysis (PCA)-based DDoS defense system, which extracts nominal traffic characteristics by analyzing intrinsic dependency across multiple attribute values. The PCA-based scheme differentiates attacking packets from legitimate ones by checking if the current traffic volume of the associated attribute value violates the intrinsic dependency of nominal traffic. The correlation among different attributes makes it more difficult for the attacker to accurately discover the statistic filtering rules and, thus, makes it highly robust to cope with new and more sophisticated attacks.

***Index Terms*—Distributed Denial-of-Service Attack, Principal Component Analysis, Statistical Filtering Rules, Selective Packet Discarding.**

I. INTRODUCTION

One of the major threats to cyber security is the Distributed Denial-of-Service (DDoS) attack in which the victim network elements are bombarded with high volumes of attacking traffic. The aim of the DDoS attack is to overload the victim and render it incapable of performing normal communications or transactions. Since the attacking traffic can be of various forms (including fictitious email messages, file transfers, http requests, as well as TCP, UDP, ICMP, and TCP-SYN packet flood with random packet attribute values), it is difficult to differentiate the attacking packets from legitimate ones. Worse still, such attacking traffic often originates from a large number of compromised machines, possibly with spoofed source IP addresses or innocent "zombie" hosts under the control of hackers. How to differentiate and block the malicious traffic without influencing legitimate traffic remain very challenging issues. Here, we further describe a sophisticated DDoS attack that seriously threatens the existing statistical filtering (or nominal traffic profile) approaches [1]-[8] and has not been solved yet.

Adaptive Attacks with statistical filtering rules Scanning (AAS): The viability of those aforementioned statistical filtering (or nominal traffic profile) approaches [1]-[8] is based on the premise that attackers do not know the victim's nominal traffic profile and cannot fake legitimate traffic. However, once the statistical filtering rules-based DDoS defense system [1]-[8] is widely deployed, DDoS attack tools may be invented to learn the statistical filtering rules of the defense system or the victim's nominal traffic profile. For instance, probing packets are crafted and sent to the victim. By observing the corresponding responses from the victim or performance degradation of the victim's network, the attacker can estimate the victim's nominal traffic profile. When attackers have successfully obtained the victim's nominal traffic profile, even just partially, they start controlling zombies to generate flooding traffic according to these nominal traffic profiles. Under this intelligent attack, those statistical filtering rules-based DDoS defense systems will fail to function. This is because attacking and legitimate packets share the same statistical property and the only thing that the defense system can do is to rate-limit traffic to the victim's bandwidth by randomly dropping packets.

In this paper, we propose a new Principal Components Analysis (PCA)-based packet filtering method to differentiate attacking packets from legitimate ones. We show that the PCA-based method is more robust than our previously proposed Conditional Legitimate Probability (CLP) and ALPi [1][2] methods when defending AAS, since its statistical filtering rules are more difficult for the attacker to discover.

This paper is organized as follows. In Section II, we describe related works in defending against DDoS attacks. In Section III, we propose a new PCA-based DDoS defense system that effectively differentiates attacking packets from legitimate ones and makes it difficult for the attacker to learn the statistical filtering rules of the defense system. In Section IV, we evaluate the PCA-based DDoS defense system by

using real Internet traffic. We conclude in Section V.

II. Related Work

Most Internet Service Providers (ISPs) rely on manual intervention for DDoS attacks. This results in poor response times and may fail to protect the victim before severe performance degradation and revenue losses are realized. Schemes proposed in [9]-[12] focus on traffic marking and traceback. A victim identifies attacking packets and reconstructs the attacking path based on the information that each router marks on the packets. However, these schemes need the ISP's cooperation, complex calculations, and long convergence times for path reconstruction. The D-WARD approach [13] detects abnormal asymmetrical behavior of two-way traffic and aims to stop DDoS attacks near the attacking sources. Unfortunately, there is little incentive for an ingress network administrator to bear the complexity and cost only to protect others' networks. Pushback mechanisms [14]-[16] have been proposed to extract attacking signatures by rate-limiting the suspicious traffic destined to the congested link. The information about ongoing attacks is propagated to upstream routers so that attacking traffic can be filtered out early. However, the effectiveness is contingent on the ability to extract a precise characterization of the attacking packets; otherwise, the legitimate traffic will be equally affected by the pushback mechanism.

There are also commercial products, e.g., Toplayer [17], which detect and mitigate specific types of known DDoS attacks, especially those generated by well-known DDoS attack tools. Arbornetworks' product [18] mitigates DDoS attacks by using a traceback approach, but requires the precise characterization of the attacking packets. Cisco (Riverhead) and Mazu's products [19] are built on statistics-based adaptive filtering techniques. However, most of these solutions do not fully automate packet differentiation or filtering. Instead, they only recommend a set of binary filter rules to the network administrator. In generally, the rule set is often too complex to be comprehensible, let alone to be debugged or modified.

Our prior work [1]-[4] and other statistical filtering rules-based approaches [5]-[8] defend against DDoS attacks by distinguishing attacking packets from legitimate ones via fine-grain traffic profile comparisons between the measured current traffic profile and the victim's nominal one. These schemes can block virtually all kinds of DDoS attacks as long as the attackers cannot precisely mimic the victim's traffic characteristics. However, the intelligent AAS attack seriously cripples those statistical filtering rules-based DDoS defense systems (as shown in Section IV).

III. A Principal Components Analysis (PCA)-based Robust DDoS Defense System

In this section, we introduce a PCA-based DDoS defense system, which extracts nominal traffic characteristics by analyzing the intrinsic dependency among each packet attribute value and differentiates suspicious traffic by analyzing abnormalities across multiple attributes. PCA is a common statistical method used in multivariate optimization problems to reduce the dimensionality of data while retaining a large fraction of the data characteristic. Since PCA only keeps the information of principal components, the nominal traffic characteristics across multiple attributes can be significantly compressed.

This scheme measures the nominal traffic and extracts nominal characteristic, when there is no DDoS attack, e.g., traffic rate is steady and much less than the protected network's bandwidth. When the congestion of the victim's access link is detected and some of the incoming packets have to be discarded, the PCA-based scheme is activated to selectively drop suspicious packets according to the packet legitimate probability and thus let more legitimated packets access the protected networks. Section A describes how to measure the nominal traffic profile in each timeslot. Section B describes how to extract the principal components from the original nominal traffic profile. Section C describes how to estimate packet legitimate probability. Section D describes how to selectively discard suspicious packets.

A. Nominal Traffic Measurement

The traffic destined to the protected networks or servers is measured at the access link to obtain an iceberg-style histogram associated with the packet's attribute. Candidate packet attribute values in the IP/TCP/UDP header considered to be used for traffic profiling include: 1) source IP prefixes, 2) Time-to-Live (TTL) values, 3) server port numbers, 4) protocol-type values, 5) packet size, 6) IP/TCP header length, and 7) TCP flag combinations, e.g., SYN, FIN, RST, ACK, and SYN-ACK.

For source IP prefixes, server port number, and packet size, attribute values that do not appear frequently (relative frequency less than a preset fixed threshold, e.g., 0.1%) during the measurement interval are grouped in a single entry in a nominal traffic histogram until the sum of their frequencies becomes higher than the threshold (0.1%). This way, the dimensions of nominal traffic data can be significantly reduced while keeping a suitable granularity for these less-frequent attribute values.

We let k denote the number of attribute values and t denote the number of successive time intervals of interest. Let X be the $k \times t$ measurement matrix, which denotes the traffic rate in packets/sec in the time series of all attribute values. Thus, each row i denotes the time series of the i-th attribute value and each column j represents an instance of all attribute values at time j. All vectors in this paper are column vectors unless otherwise noted. All matrices are denoted by uppercase letters and vectors are denoted by lowercase letters.

B. PCA of Nominal Traffic

PCA is a coordinate transformation method that maps a given set of data points onto new axes. By variance maximizing rotation of the original variable space, PCA reduces the amount of dimensions required to classify the given set of data points and produces a set of principal

components, which are orthogonal eigenvectors [20]. In other words, PCA projects a new set of axes that best suit the data. By using the new axes, the given set of data points can be described with minimal storage requirements.

The data elements used to describe the nominal traffic characteristics associated with packet attribute values are high, e.g., 1000, even though those less-frequent attribute values are already grouped into one entry. Generally, those high dimension data elements have dependencies or correlations among the different attribute values. For example, there are intrinsic dependencies between such particular attribute value pairs as source IP prefixes and TTL values, source IP prefixes and server port numbers, protocol-type values and server port numbers, and packet size and TCP flag combinations. Those intrinsic dependencies or correlation structures among the data elements can thus be exploited by PCA to analyze the anomalous behavior by discovering the data elements that violate the normal dependency structure [21][22], and reduce the storage requirement for describing nominal traffic characteristics.

After obtaining a nominal traffic measurement matrix X, we let m denote the mean vector, which is calculated by taking the mean of each attribute value over time. Let X_z denote the zero-mean measurement matrix, in which, for each column j $(1 \le j \le t)$, $x_{zj} = x_j - m$. We let C denote the covariance matrix, which quantifies the correlation between every attribute value pair and can be written as:

$$C = \left(X_z X_z^T\right) / (t-1) \quad (1)$$

In the covariance matrix, C_{ij} is the covariance between attribute value i and j. There are k eigenvalue/eigenvector pairs for the covariance matrix. Let e_i and v_i denote the i-th $(1 \le i \le k)$ eigenvalue and eigenvector of covariance matrix C, respectively. The eigenvalue e_i of the eigenvector v_i corresponds to the relative amount of variance it encompasses in the direction of v_i. The larger the eigenvalue, the more significant its corresponding projected eigenvector is [23]. The eigenvectors are sorted from most to least significance, based on its corresponding eigenvalues. The eigenvector with the most-significant eigenvalue is the first principal component, which has the property that it points in the direction of maximum variance and can be expressed as:

$$v_1 = arg \max_v \left(v^T X_z X_z^T v\right), \ s.t. \ v^T v = 1 \quad (2)$$

The next principal components each capture the maximum variance among the remaining orthogonal directions. Here, we let X_{zq}^T denote the remaining matrix, which eliminates the projection of mapping X_z^T onto the first $q-1$ $(2 \le q \le t)$ principal components.

$$X_{zq}^T = X_z^T - \sum_{i=1}^{q-1} X_z^T v v^T \quad (3)$$

Proceeding iteratively, once the first $q-1$ principal components have been determined, the q-th principal component captures the maximum variance among the remaining orthogonal axis and can be written as:

$$v_q = arg \max_v \left(v^T X_{zq} X_{zq}^T v\right), \ s.t. \ v^T v = 1 \quad (4)$$

Thus, the principal components are ordered by the amount of data variance that they capture. Here, we let P denote the principal components matrix, which includes a set of eigenvectors $[v_1, v_2, ..., v_n]$ corresponding to the n most-significant eigenvalues of covariance matrix C. For the remaining eigenvectors with eigenvalues that are close to zero or very small, we can essentially discard them to reduce the dimensionality of the new basis.

C. Attribute Value Legitimate Probability Estimation

After abstracting the principal components, we use the intrinsic dependency structure among multiple attribute values to quantify the anomaly for each attribute value by discovering data points that violate the normal dependency structure. Since the principal components capture the most variance (e.g., 99% of the total) of the nominal traffic data, the data points of legitimate traffic rate measured according to the attribute value should be close to those principal components. For example, there is nominal traffic from source IP prefix 128.238, with TTL value 56 to the server http://www.yahoo.com. If the rate of this kind of nominal traffic increases, we can observe that the traffic rate associated with source IP prefix 128.238 and TTL value 56 at the edge router of yahoo increases simultaneously and vice versa. So there is intrinsic dependency between the attribute value source IP prefix 128.238 and TTL value 56. The nominal traffic measurement points without a DDoS attack is usually close to the extracted principal component, shown by the grey dots in Fig. 1.

Because attacking traffic doesn't follow the correlation of nominal traffic, for instance, the DDoS attacking traffic using 128.238 as a spoofing source IP prefix with random TTL values, the current traffic measurement points deviate from those principal components, as shown by the dark dot in Fig. 1. The more attacking traffic that comes in, the greater the deviation.

Here, we let y denote the current traffic measurement vector, which measures the traffic rate destined to the protected end point associated with each attribute value at the current interval. Let y_l and y_a denote the current legitimate and attacking traffic estimation vector, respectively. $(y = y_l + y_a)$

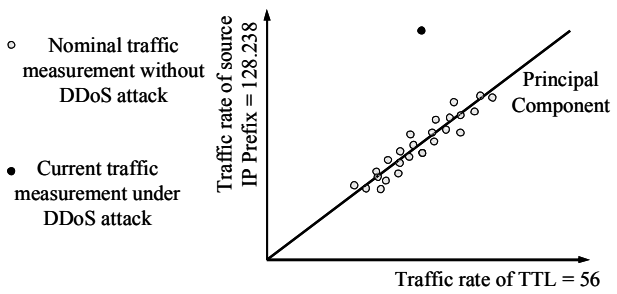

Fig. 1. Correlation among multiple attribute values.

Since DDoS attackers try to launch as much attacking traffic as possible, the traffic volume of the attribute value shared with attacking traffic increases significantly. On the other hand, since the attacker usually does not know the target's nominal traffic profile to mimic the legitimate traffic characteristics, the intrinsic dependency across multiple attribute values in the nominal traffic will be violated. Our proposed scheme tries to filter out the suspicious traffic that does not obey the intrinsic dependency of nominal traffic and has an abnormally huge traffic volume. Based on the above observation, we transform the problem of how to estimate the legitimate probability for each attribute value to an optimal problem, which can be expressed as follows.

$$y_l = arg \max_{y_l} \|y_l\|_1, \quad s.t. \begin{matrix} 0 \le y_{li} \le y_i, 1 \le i \le k \\ PP^T y_l = y_l \end{matrix} \qquad (5)$$

in which, $\|y_l\|_1 = \sum_i |y_{li}|$ is the sum of the absolute value of each element of vector y_l. The objective of (5) is to find a solution to vector y_l to maximize the legitimate likelihood of current incoming packets with the following constraints. First, any element in legitimated traffic vector y_l should not be less than zero and larger than the corresponding element in the current traffic vector y. Second, the point y_l is on the space of principal components ($PP^T y_l = y_l$), which in based on the observation that legitimate traffic vectors are usually very close to the principal components. Considering that the traffic volume associated with each attribute value is not static, we modify y as $max(y, y_{mean}+3\sigma)$ in which, y_{mean} and σ are the average and standard deviation vector of the nominal traffic rate associated with each attribute value.

(5) is a standard problem of a linear program that can be efficiently solved by the Simplex method. However, the k (k is around 1000) unknown variables and $3k$ constraints cause a formidable computing requirement. We transfer (5) to an equivalent that can be written as:

$$y_l = arg \max_{Ps} \|Ps\|_1, \quad s.t. \ 0 \le Ps \le y \qquad (6)$$

in which, s is a projection vector ($n \times 1$) that quantifies the distance from the original point to point y_l along each principal axis. It can be verified that $P^T y_l = s$ or $y_l = Ps$. In (6), there are only n (n is much smaller than k, e.g., 30, when the set of principal components captures 99% of total variance) unknown variables and $2k$ constraints. In our simulation, the Simplex method is remarkably fast in solving (6), in general, less than 2 seconds using a desktop with a 3-GHz CPU. Finally, for attribute value i, the legitimate probability (or partial score) is defined as y_{li}/y_i, e.g., 1.2kpps/2kpps = 60% for the attribute value source IP prefix 12.238 as shown in Fig. 2. For an incoming packet, we assign a packet score by multiplying the partial score of each attribute value that packet possesses.

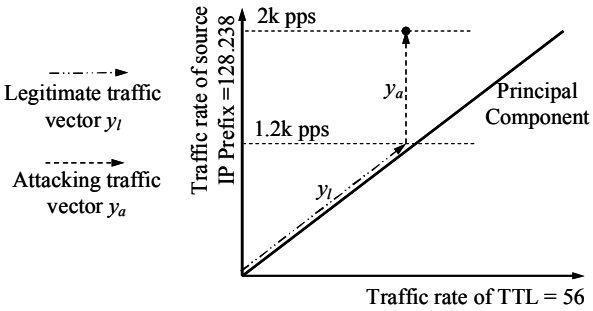

Fig. 2. Packet legitimate probability estimation.

D. Packet Legitimate Probability (PLP) Estimation

When a packet comes in, the legitimate probability for each attribute value that it processes is obtained by looking up the scorebook. The packet legitimate probability (PLP) then can be calculated by multiplying the legitimate probability for each attribute value together. Once the PLP is computed for each suspicious packet, selective packet discarding and overload control can be conducted using PLP as the differentiating metric. The key idea is to prioritize packets based on their PLP values. We maintain the CDF (Cumulative Distribution Function) of the PLP of all incoming suspicious packets using one-pass quantile computation techniques similar to those described in [24][25]. We then discard a suspicious packet if its PLP value is below a dynamically adjusted threshold, which is looked up from a previous snapshot of the CDF of the PLP values of all suspicious packets. The adjustment of the PLP discarding threshold, as well as the load-shedding algorithm, is expected to operate at a time-scale that is considerably longer than the packet arrival time-scale.

IV. Performance Evaluation

In this section, we evaluate the performance of the proposed PCA-based packet filtering scheme in a standalone setting via simulation and compare it with our previous work, the CLP-based packet filtering scheme. We use the same Internet trace archive from the WIDE project [26] and simulation setting described in Section A. In Section B, we evaluate both PCA- and CLP-based methods when defending against various static and dynamic attacks including: random attack, SYN flood attack, SQL worm slammer attack, nominal attack, mixed attack, and dynamic attack. In Section C, we show that the PCA-based DDoS defense system significantly reduces the false positive rate when defending against adaptive DDoS attacks with statistical filtering rule scanning, making the defense system more robust.

A. Simulation Setting, Performance Criteria

1) Simulation Setting

To establish a fair comparison, all simulations have the same overall common internal settings, input traffic, and attacking traffic (generated in the exact same way). Unless stated otherwise, the default settings for the simulation are summarized as follows.

We used the Internet trace archive of the WIDE project

[26]. We built the nominal traffic profile by using the trace from 1:00 pm to 1:15 pm on March 3, 2006 and used the trace after 1:15 pm on March 3 as legitimate traffic. The data rate of the legitimate traffic is about 9,500 pps, the victim's targeted rate is 20,000 pps, and the attacking traffic rate is 10 times the legitimate traffic rate.

The defense schemes can be classified into CLP- and PCA-based packet filtering approaches.

CLP: assigns a score for each arriving packet based on the so-called Conditional Legitimate Probability (CLP) [1], which is calculated based on the ratio between the nominal traffic profile and the measured traffic profile under attack.

PCA: extracts the most-significant eigenvectors as the set of principal components, which captures 99% of the total variance in the nominal traffic data, and estimates the legitimate probability for each incoming packet.

These two schemes pass or discard the packet if its legitimate probability is above a threshold or below it. The threshold is dynamically changed so as to regulate the aggregated data rate to the victim's targeted data rate. Both PCA- and CLP-based schemes build the current traffic profile and generate a scorebook for calculating each packet's legitimate probability every 20 sec.

2) Performance Criteria

We adopted the following four items as performance criteria for the simulations: false positive ratio, false negative ratio, score differentiation power, and effectiveness of the overload control. The first two are the most important. False positives (negatives) represent the percentage of legitimate (attacking) packets that are mistakenly discarded (admitted).

The score differentiation power quantifies the differences in the score distribution for attack and legitimate packets. Such differences are quantified using two metrics, namely, R_A and R_L as illustrated in Fig. 4 in [1]. Let Min_L (Max_A) be the lowest (highest) score observed for the incoming legitimate (attacking) packets. Define R_A (R_L) to be the fraction of attacking (legitimate) packets that have a score below Min_L (above Max_A). The closer the value of R_A and R_L to 100%, the better the score differentiation power is. In practice, the score distributions usually have long but thin tails due to very few outlier packets with extreme scores. To avoid the masking effect of such outliers, we have taken Min_L (Max_A) to be the 1st (99th) percentile of the score distribution of legitimate (attacking) packets.

To evaluate the effectiveness of the overload controls, we compare the actual output traffic rate R_{out} against the target maximum traffic rate R_{target}. Ideally, the R_{out}/R_{target} ratio should be very close to one (either above or below).

B. Defend against Static and Dynamic Attacks

We have evaluated and compared the performance of the aforementioned DDoS defense schemes against the following types of static and dynamic attacks:

1) *Random attack*: all attribute values of the attacking packets are uniformly randomized over their corresponding allowable ranges;
2) *SQL Slammer Worm attack*;
3) *TCP-SYN Flood attack*;
4) *Nominal attack*: all attacking packets resemble the most dominant type of legitimate packets observed in practice, i.e., 1500-byte TCP packets with server port 80, TCP flag set to ACK, and uniformly random source IP addresses;
5) *Mixed attack*: equally combines the above four types of attacks while keeping the overall attack rate to ten times that of the legitimate traffic rate;
6) *Dynamic attack*: similar to the mixed attack except that the above four types of attacks are randomly selected and continue for an exponentially distributed period.

Although random attack randomizes the attribute values in flooding traffic to disguise the attacking signature or characteristic, and nominal attack mimics the dominant type of legitimate traffic observed in practice, our proposed PCA-based packet-filtering scheme effectively filters out attacking traffic without significant impact on legitimate traffic.

From the simulation result shown in Table I, we see that the performance of the PCA-based scheme is slightly worse than that of the CLP-based scheme, when defending against random, SQL slammer worm, TCP-SYN flood, nominal, and mixed attacks. In all static attacks except the mixed attack, PCA-based scheme can distinguish attacking packets from legitimate ones with a false positive less than 0.56%, R_A and R_L above 97.5%, and the ratio R_{out}/R_{target} less than 1.06. Even in defending against mixed attack, the proposed PCA-based scheme can still successfully discard more than 87.1% of the attacking packets (together with about 1.23% of legitimate ones).

Dynamic attacks are more challenging due to their complex/ time-varying attacking packet characteristics. In our simulation, the measurement/scorebook generation interval of the defense schemes is 20 sec and the average change-period of a dynamic attack is either 20 or 60 sec. When the average change-period of the attack is much longer than the measurement/scorebook generation interval (60 sec vs. 20 sec), the change in attacking packet characteristics can readily be tracked as shown in Table I. However, when such changes occur at the same (or shorter) time-scale of the measurement update interval, the aforementioned PCA-/CLP-based schemes can be misled to defend against some no-longer-existing attacking packets and thus cause a high rate of false positives. A possible remedy is to shorten the measurement update interval or apply more sophisticated change-detection techniques [28] on the current profile measurements to trigger and speed up scorebooks/CDF updates [29]. However, even in the worst case, the proposed PCA-based schemes can still successfully discard more than 88.3% of the attacking packets (together with about 4.61% legitimate ones). This is substantially better than random packet dropping as the aggregate arrival rate is much more than the target's traffic rate. As shown in Table I, PCA-based method is more robust than CLP-based method to defend against dynamic attack.

TABLE I PERFORMANCE OF PCA/CLP BASED SCHEMES AGAINST VARIOUS STATIC AND DYNAMIC ATTACKS

Type of Attack	Type of Defense	% False +ve	% False -ve	% R_L	% R_A	$\frac{R_{out}}{R_{target}}$
Random attack	CLP	0	10.45	98.7	99.5	1.00
	PCA	0.56	10.82	97.5	97.7	1.02
SQL Slammer Worm	CLP	0	10.44	100	100	1.00
	PCA	0.01	11.70	99.9	100	1.06
TCP-SYN Flood	CLP	0	10.40	100	100	1.00
	PCA	0.09	10.77	99.0	98.8	1.02
Nominal attack	CLP	0	10.43	100	100	1.00
	PCA	0.01	10.70	100	99.9	1.06
Mixed attack	CLP	0.16	12.31	99.9	99.2	1.10
	PCA	1.23	12.81	96.9[1]	96.6[1]	1.12
Dynamic attack (60 sec)	CLP	1.14	10.38	96.7	94.4	1.00
	PCA	2.43	11.25	91.8	93.9	1.04
Dynamic attack (20 sec)	CLP	6.08	10.95	82.4[1]	86.7[1]	1.01
	PCA	4.61	11.66	89.8[1]	92.4[1]	1.05

C. Defend against Adaptive Attack with statistical filtering rules Scanning (AAS)

In this section, we applied Adaptive Attack with statistical filtering rules Scanning (AAS) and profile attack to evaluate our PCA- and CLP-based defense schemes.

1) *Profile attack*: all attributes except the source IP prefix of attacking packets are proportionally distributed over the nominal traffic profile obtained from a 10 minutes Abilene-I data set [27] starting from 9:00 AM, August 14, 2002.
2) *AAS*: all attributes of attacking packets are uniformly distributed over the scanning profile, which is obtained with statistical filtering rules scanning.

There is ample Internet trace data that is public and posted for research purposes. Open Internet trace data usually shares some common characteristics with the target's nominal traffic profile and thus can be utilized by attackers. So we launch the profile attack to evaluate the robustness of our proposed PCA-based scheme.

In AAS, statistical filtering rules scanning initially sends probing traffic of each testing attribute value at a given rate (e.g., 100 pps). The rules scanning exponentially increases the probing traffic rate based on observing if most probing packets were passed. In simulation, we let AAS scan the attribute packet size and server port number only in the range from 0 to 1500 Bytes, and from 0 to 1024, respectively. This is because most packets fall into those two ranges. The attacker can significantly reduce the scanning periods and get a more accurate scanning profile by just scanning in those ranges.

The PCA- and CLP-based defense systems build the current traffic profile and update the local database for calculating each packet's score every 20 sec. We scan the nominal traffic profile with the attribute TTL value, packet size, protocol type, TCP flag, and server port number. To quickly reach the dropping point for each attribute value, we exponentially increase the launching rate for each testing attribute value until we observe the defense system dropping most of those probing packets. We keep the dropping point for each possible attribute value to obtain the nominal traffic profile. An example of the scanning results for the TTL value and the corresponding nominal traffic profile for the CLP-based method are shown in Fig. 4 and 3, respectively. We can see that the attacker can successfully learn the majority of the CLP-based nominal traffic profile, even though there are some scanning errors caused by variations of legitimate or random flooding traffic. After an attacker successfully obtains the target's nominal traffic profile, the attacker launches attacking traffic according to the scanning results.

In Table II, we show the performance of the PCA- and CLP-based schemes against profile and AAS attacks. For the CLP-based scheme, the number of false positives under profile and AAS attacks is much larger than that of the static random attack, in which the attacker does not have any knowledge of the nominal profile and generates attacking traffic with random attribute values. In other words, the attacker really obtains some useful information about the target's nominal traffic profile by using the profile and statistical filtering rules scanning. You can see from Table II that the AAS dramatically increases the false positive to 17.85% and profile attack causes 7.81% false positives.

Although the false positive rate of the PCA-based method is slightly worse than that of the CLP scheme when defending against various static attack (shown in Table I), we can see that our proposed PCA-based scheme is much more robust than the CLP-based scheme from Table II. The false positive rate of PCA is 2.69% and 2.71%, compared to 7.81% and 17.85 of the CLP-based scheme, when defending against profile attack and AAS, respectively. Even in the worst case, our proposed PCA-based scheme can still successfully discard more than 89% of the attacking packets (together with about 2.71% of the legitimate ones). This is substantially better than using the CLP-based scheme to defend against profile attack and AAS. The time for statistical filtering rules scanning is 249 and 255 period for CLP- and PCA-based scheme, respectively. Each period is 20 sec.

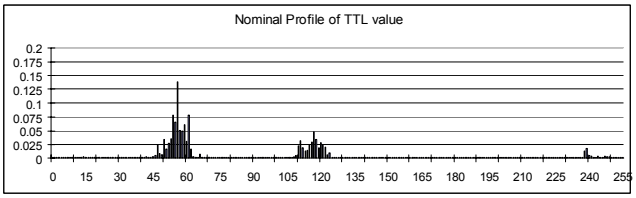

Fig. 3. Nominal traffic profile of TTL value for CLP-based scheme.

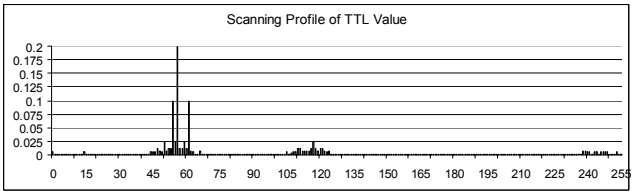

Fig. 4. Scanning profile of TTL value by AAS for CLP-based scheme.

[1] Instead of the 1st (99th) percentile, we take Min_L (Max_A) to be the 5th (95th) percentile of the score distribution of legitimate (attacking) packets.

TABLE II PERFORMANCE OF PCA/CLP-BASED SCHEMES AGAINST PROFILE ATTACKS AND AAS

Type of attack	Type of defense	% False +ve	% False -ve	$\frac{R_{out}}{R_{target}}$	Period
Profile attack	CLP	7.81	11.20	1.00	0
	PCA	2.69	10.87	1.00	0
AAS	CLP	17.85	12.40	1.01	249
	PCA	2.71	10.92	1.01	255

The PCA-based scheme differentiates suspicious packets by analyzing abnormalities across multiple attribute values. Generally, those multiple attribute values have intrinsic dependencies or correlations that are quantified by principal components. When statistical filtering rules scanning changes the probing traffic rate of a particular testing attribute value, the dependency among those related attribute values is broken. As a result, it is difficult for statistical filtering rules scanning to obtain certain important information about the nominal traffic profile, unless statistical filtering rules scanning tries to learn the nominal traffic profile with joint attribute. However, the huge amount of possible joint attribute values makes such joint attribute value-based statistical filtering rules scanning almost impossible.

V. CONCLUSION

In this paper, we first presented the vulnerability of those statistical filtering rules-based DDoS defense systems when defending against future adaptive attacks with statistical filtering rules scanning. Our simulation result shows that this kind of intelligent adaptive attack seriously cripples the performance of our previous work, PacketScore, a statistical filtering rules-based scheme.

We propose a PCA-based robust DDoS defense system, which extracts nominal traffic characteristics by analyzing the intrinsic dependency across multiple attribute values. This PCA-based scheme differentiates attacking packets from legitimate ones by checking whether the current traffic volume of associated attribute values violates the intrinsic dependency of nominal traffic. The correlation among different attributes makes it difficult for attackers to accurately discover the statistic filtering rules, and thus makes it highly robust to cope with new and more sophisticated attacks in the future. Our results show that our proposed PCA-based scheme effectively differentiates attacking packets from legitimate ones, when defending against various static, dynamic, and adaptive attacks.

REFERENCES

[1] Y. Kim, W. Lau, M. Chuah, J. Chao, "PacketScore: Statistical-based Overload Control against Distributed Denial-of-Service Attacks," In *Proceedings of Infocom*, 2004.

[2] P. Ayres, H. Sun, H. Chao, "ALPi: A DDoS Defense System for High-Speed Networks," *IEEE J-SAC high-speed network security – architecture, algorithms, and implementation,* 2006.

[3] P. Ayres, H. Sun, H. Chao, Wing C. Lau, "A Distributed Denial-of-Service Defense System Using Leaky-Bucket-Based PacketScore," *ACNS 2005,* New York, June 2005.

[4] M. Chuah, Y. Kim, W. Lau, J. Chao, "Transient Behavior of PacketScore," In *Proceedings of ICC*, 2004.

[5] Q Li, EC Chang, MC Chan, "On the Effectiveness of DDoS Attacks on Statistical Filtering," In *Proc. IEEE INFOCOM*, 2005.

[6] C. Jin, H. Wang, K. G. Shin, "Hop-count filtering: An effective defense against spoofed DDoS traffic," In *Proc. 10th ACMConf. Comput. Commun. Security,* Oct. 2003, pp. 30–41.

[7] C.C. Zou, N. Duffield, D. Towsley, W. Gong, "Adaptive Defense Against Various Network Attacks," In *IEEE J-SAC high-speed network security –architecture, algorithms, and implementation*, Oct. 2006.

[8] L. Kencl, C. Schwarzer, "Traffic-Adaptive Packet Filtering of Denial of Service Attacks," In *Proceedings of the 2006 International Symposium on WOWMOM,* June 2006.

[9] Y. Xiang, W. Zhou, Z. Li, and Q. Zeng, "On the Effectiveness of Flexible Deterministic Packet Marking for DDoS Defense," In *IFIP International Workshop on NSS* 2007.

[10] Z. Gao, and N. Ansari, "Enhanced probabilistic packet marking for IP traceback," In *Proceedings of GLOBECOM*, 2005.

[11] Jing, Y. Wang, X. Xiao, X. Zhang, and Gendu, "NIS04-5: Defending Against Meek DDoS Attacks By IP Traceback-based Rate Limiting," In *Proceedings of GLOBECOM* 2006.

[12] M. Sung and J. Xu, "IP Traceback-based Intelligent Packet Filtering: A Novel Technique for Defending Against Internet DDoS Attacks," In *Proc. of IEEE Transactions on Parallel and Distributed Systems*, Vol. 14, No. 9, pp. 861-872, Sep 2003.

[13] J. Mirkovic, G. Prier, and P. Reiher, "Attacking DDoS at the Source," In *Proceedings of 10th IEEE International Conference on Network Protocols*, November 2002.

[14] J. Ioannidis and S.M. Bellovin, "Implementing Pushback: Router-Based Defense against DDoS Attacks," In *Proceedings of Network and Distributed System Security Symposium*, February 2002.

[15] R. Chen and J-M. Park, "Attack diagnosis: throttling distributed denial-of-service attacks close to the attack sources," *In Proceedings of the 14th International Conference on Computer Communications and Networks*, Oct. 2005.

[16] M. Cai, Y. Chen, Y.-K. Kwok, and K. Hwang, "A Scalable Set-Union Counting Approach to Pushing Back DDoS Attacks," USC *GridSec Technical Report TR-2004-21*, Oct. 2004.

[17] Toplayer Inc., http://www.toplayer.com

[18] Arbor Networks Inc., http://www.arbornetworks.com

[19] Mazu Networks Inc., http://www.mazunetworks.com

[20] Jolliffe. I. T. Principal Component Analysis. Springer-Verlag, NY, 2002

[21] Y. Zhang, Z. Ge, M. Roughan, and A. Greenberg. "Network anomography," In *Proceedings of ACM/USENIX Internet Measurement Conference (IMC)*, 2005.

[22] A. Lakhina, M. Crovella, and C. Diot. "Diagnosing network-wide traffic anomalies," In *Proc. ACM SIGCOMM 2004*, August 2004.

[23] D. Nguyen, A. Das, G. Memik, and A. Choudhary, "A Reconfigurable Architecture for Network Intrusion Detection using Principal Component Analysis," In *Proceedings of the 2006 ACM/SIGDA*, 2006.

[24] F. Chen, D. Lambert and J. C. Pinheiro, "Incremental Quantile Estimation for Massive Tracking," In *Proceedings of the 6th International Conference in Knowledge Discovery and Data Mining*, August 2000.

[25] A.C. Gilbert, Y. Kotidis, S. Muthukrishnan, and M. J. Strauss, "How to Summarize the Universe: Dynamic Maintenance of Quantiles," in *Proceedings of the 28th VLDB Conference*, August 2002.

[26] http://mawi.wide.ad.jp/mawi/samplepoint-B/2006/

[27] http://pma.nlanr.net/Traces/long/ipls1.html

[28] S. Kasera, J. Pinheiro, C. Loader, M. Karaul, A. Hari, and T. LaPorta, "Fast and Robust Signaling Overload Control," In *Proceedings of 9th International Conference on Network Protocols (ICNP),* November 2001.

[29] F. Keresteciogl, *Change detection and input design in dynamical systems*, John Wiley 1993.