Monitoring the Macroscopic Effect of DDoS Flooding Attacks

Jian Yuan Kevin Mills

Advanced Network Technologies Division

National Institute of Standards and Technology

February 4, 2004

Abstract: Creating defenses against flooding-based, distributed denial-of-service (DDoS) attacks requires real-time monitoring of network-wide traffic to obtain timely and significant information. Unfortunately, continuously monitoring network-wide traffic for suspicious activities presents difficult challenges because attacks may arise anywhere at any time, and because attackers constantly modify attack dynamics to evade detection. In this paper, we propose an efficient method for early attack detection. Using only a few observation points, our proposed method can monitor the macroscopic effect of DDoS flooding attacks. We show that such macroscopic-level monitoring might be used to capture shifts in spatial-temporal traffic patterns caused by various DDoS attacks, and then to inform more detailed detection systems about where and when a DDoS attack probably arises in transit or source networks. We also show that such monitoring enables DDoS attack detection without any traffic observation in the victim network.

Keywords: DDoS attack, monitoring, network traffic, attack dynamics, spatial-temporal pattern

1 Introduction

The success of the Internet can be attributed in large part to the end-to-end principle, which enabled deploying a relatively simple network infrastructure (i.e., packet-forwarding nodes supported by a few routing protocols), allowing network applications to evolve independent of the core network [1]. In particular, the end-to-end congestion control mechanisms of the TCP (Transmission-Control Protocol) have played a key role, both in illustrating the end-to-end principle and in achieving a robust and stable Internet. At the same time, the existing end-to-end mechanisms have been proven highly ineffective at protecting the Internet from distributed denial-of-service (DDoS) attacks, which have become an increasingly frequent, global disturbance [2].

A DDoS attack can be characterized as a simultaneous network attack on a victim (e.g., a web server or a router) from a large number of compromised hosts, which may be distributed widely among different, independent networks. By simply exploiting the tremendous asymmetry existing between network-wide resources and local capacities of a victim, a flooding-based DDoS attack can build up an intended congestion very quickly at an attacked target. The Internet routing infrastructure, which is stateless and based mainly on destination addresses, appears extremely vulnerable to such large-scale, coordinated attacks. DDoS attacks cannot be detected and stopped easily because forged source addresses and other sophisticated techniques are used to conceal attack sources. DDoS flooding attacks can take a victim network off the Internet even without exploiting particular vulnerabilities in network protocols or weaknesses in system design, implementation, or configuration. While applying security patches may avert attacks against protocol or system vulnerabilities, congestion-inducing DDoS attacks exploit an inherent

weakness in the Internet design, and thus present a serious threat to Internet stability. This paper focuses on such flooding-based DDoS attacks.

In general, packet filtering embodies the essential response to confirmed DDoS attacks. However, an ability to detect attacks directly affects the ability to filter DDoS flows accurately and to alleviate damage. There have been some efforts to deploy detection mechanisms at the attack victim, in transit networks, and in networks containing attack sources [3]. Usually, flooding attacks can be most easily detected at the victim network, where all the generated attack packets can be observed. Unfortunately, an attack victim cannot defeat a flooding attack simply through detection. Instead, attack packets must be filtered in transit networks, preferably close to attack sources, before they converge on the victim network. Unfortunately, attempts in transit networks to detect such attacks, as anomalies in voluminous and aggregated traffic observed at core routers, do not often succeed without a high false-alarm rate. Similarly, networks hosting attack sources cannot readily achieve effective attack detection [4]. When a DDoS attack arises, a network with attack sources may observe only a normal outgoing load. In fact, studies characterizing Internet traffic show high variability in legitimate connections [5]. Use of attack-obfuscation techniques further increases the difficulty of detecting attacks with high confidence in a short period. Where sophisticated attacks (such as increasing rate, fluctuating rate, and natural-network-congestion-like attacks [3, 6]) are used, tension between accuracy and detection latency increases the difficulty of successful, real-time detection.

Although flooding attacks might successfully avoid various detection techniques by changing attack signatures, the effect of attacks will remain unchanged: congestion will be induced by attack traffic at the victim network. Can one exploit this fact to improve detection performance in transit or source networks? In this paper, we argue that DDoS attacks alter the

internal characteristics of network-wide traffic so that an appropriate monitoring method can accurately detect large-scale flooding attacks without observing traffic in the victim network.

To avoid congestion in the Internet, all flows under end-to-end controls adapt themselves in a self-organized, distributed manner. Adaptive behaviors of flows in different directions play a crucial role to keep the Internet stable and to form macroscopic traffic patterns. During a DDoS attack, the attack packets do not observe end-to-end congestion control algorithms; rather, they overwhelm the intended victim, causing well-behaved flows to back off and then ultimately to starve. Large-scale DDoS attacks also impair other traffic flows that happen to share a portion of the congested network. Such network-wide phenomena might show themselves in shifting patterns of spatial-temporal traffic, which can be captured, e.g., by using a novel technique we recently developed for analyzing macroscopic behavior [7]. Our analysis technique can infer the congestion state of specific network areas without directly measuring them. This trait proves advantageous in DDoS-attack detection because congested routers near the victim may fail to collect and transfer measurement data.

There is a high cost for continuously monitoring network-wide traffic for various suspicious activities because attacks can emerge anywhere at any time. In addition, suspicious activities do not always relate to DDoS attacks. For example, flash crowds on the Internet can trigger false alarms in detection systems. Also, sophisticated DDoS attacks may mimic natural network congestion [6]. Therefore, it would be desirable if attack-detection algorithms could be activated with a narrowed focus in response to an alarm raised by an efficient, continuous monitoring operation. In this paper, we show that the macroscopic behavior of large-scale networks could provide significant information to defend against serious DDoS attacks, which exhibit apparent effects on network congestion even while attackers constantly modify their techniques to avoid

detection. Coupled with a dynamic monitoring capability deployed in transit or source networks, our analysis method could provide a promising guidance function to warn detection systems about where and when a DDoS attack probably arises.

Since DDoS-attack patterns are becoming more sophisticated and effective, we experiment in this paper with different attack modes: constant rate, increasing rate, natural-network-congestion-like, subgroup, pulsing, and TCP-targeted attacks. We use simulation results to show how our macroscopic-level technique monitors spatial-temporal patterns under diverse DDoS flooding attacks in a large-scale network. We show that these attacks, which have an apparent effect on network congestion, reveal themselves in shifts of spatial-temporal traffic patterns without any observations from the suffering victim. The dynamic nature of some attacks may even become an advantage because our technique benefits from increased correlation arising under shifting patterns in network traffic. The rest of this paper is structured as four sections. Section 2 explains our technique to analyze macroscopic behavior. In Section 3, we describe our simulation model. In Section 4, we report and discuss our simulation results. We present concluding remarks in Section 5.

2 Analysis Technique

In large-scale networks, such as the Internet, spatial-temporal correlations emerge from interactions among adaptive transport connections and from variations in user demands. Capturing the macroscopic patterns of the correlations over time can help us to understand shifting traffic patterns, to identify operating conditions, and to reveal traffic anomalies. Motivated by insights about network dynamics at the macroscopic level, we propose a novel method to analyze spatial-temporal traffic at large scale [7]. In particular, by exploiting the

property of increased correlation, our analysis method can efficiently infer a shift in the spatial-temporal traffic pattern of large areas of interest outside those few areas where measurements are made. More details about this method can be found elsewhere [7]. In this paper, we apply our analysis method to watch network-wide patterns based on measurements from only a few observation points.

2.1 Representing network flow data

Assume that there are N subnets, interconnecting through backbone routers to form a large-scale distributed network, where L subnet routers are deployed as observation points to log outbound traffic. First, we need to represent packets flowing between distinct source-destination pairs at each sampling interval. Let $\boldsymbol{x} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_N})^T$ denote the flow vector of corresponding packet counts, observed in L subnets during a given time interval. Each element of this flow vector is itself a vector defining the number of packets flowing into the corresponding subnet from each of the observation subnets in the distributed network. The method to obtain all flow variables in this vector is to first enumerate all the destination subnets and then the observation posts by 1 to L, and group these indices by subnet: the subnets sending to the first subnet in the first block, $\boldsymbol{x_1}$, and those sending to the second subnet in the second block, $\boldsymbol{x_2}$, and so forth. Thus, we form $\boldsymbol{x}$ with subvectors in the order $\boldsymbol{x_1} = (x_{11}, x_{21}, \ldots, x_{L1})^T$, $\boldsymbol{x_2} = (x_{12}, x_{22}, \ldots, x_{L2})^T$, …, $\boldsymbol{x_N} = (x_{1N}, x_{2N}, \ldots, x_{LN})^T$, where x_{ij} represents packet flow from the ith observation point (i =1, 2, …, L) to the jth subnet (j =1, 2, …, N). Each flow variable x_{ij} is normalized as f_{ij} by its mean m_{ij} and standard deviation σ_{ij},

$$f_{ij} = (x_{ij} - m_{ij}) / \sigma_{ij}. \qquad (1)$$

Then, the normalized flow vector $\boldsymbol{f}$, corresponding to $\boldsymbol{x}$, comprises N normalized subvectors, $\boldsymbol{f}_k$ (k =1, 2, …, N), where each subvector is formed from normalized flow variables f_{ik} ($i \le L$ and $k \le N$).

2.2 Cross-correlation analysis

Cross-correlation analysis is a tool commonly used to analyze multiple time series. We can compute the equal-time cross-correlation matrix $\boldsymbol{C}$ with elements

$$C_{(ij)(kl)} = \left\langle f_{ij}(t) f_{kl}(t) \right\rangle, \tag{2}$$

which measures the correlation between f_{ij} and f_{kl}, where $\langle \cdots \rangle$ denotes a time average over the period studied. The cross-correlation matrix is real and symmetric, with each element falling between –1 and 1. Positive values indicate positive correlation, while negative values indicate an inverse correlation. A zero value denotes lack of correlation.

We can further analyze the correlation matrix $\boldsymbol{C}$ through eigenanalysis [8]. The equation

$$\boldsymbol{C}\boldsymbol{w} = \lambda \boldsymbol{w} \tag{3}$$

defines eigenvalues and eigenvectors, where λ is a scalar, called the eigenvalue. If $\boldsymbol{C}$ is a square K-by-K matrix, e.g., $K = L(N-1)$ here, then $\boldsymbol{w}$ is the eigenvector, a nonzero K by 1 vector (a column vector). Eigenvalues and eigenvectors always come in pairs that correspond to each other. This eigenvalue problem has K real eigenvalues, some of which may repeat. An eigenvector is a special kind of vector for the matrix it is associated with, because the action of the underlying operator represented by the matrix takes a particularly simple form on the eigenvector input: namely, simple rescaling by a real number multiple. The eigenvector $\boldsymbol{w}^1$ corresponding to the largest eigenvalue λ_1 often has special significance for many applications.

There are various algorithms for the computation of eigenvalues and eigenvectors [8]. Here, we exploit the MATLAB **eig** command, which uses the QR algorithm to obtain solutions [9].

2.3 Defining the weight vector

The cross-correlation matrix contains within itself information about underlying interactions among various flows. The components of the eigenvector $\boldsymbol{w^1}$ of the largest eigenvalue λ_1 represent the corresponding flows' influences on macroscopic behavior, abstracted from the matrix $\boldsymbol{C}$ into the pair (λ_1, $\boldsymbol{w^1}$). The eigenvector $\boldsymbol{w^1}$ comprises N subvectors, i.e., $\boldsymbol{w^1} = (\boldsymbol{w^1_1}, \boldsymbol{w^1_2}, \ldots, \boldsymbol{w^1_N})^T$. The kth subvector $\boldsymbol{w^1_k}$, corresponding to the kth subnet, is formed from components w^1_{ik} ($i \le L$ and $k \le N$) representing the ith obsevation point's contribution to the kth subnet. We consider the square of each component, $(w^1_{ik})^2$, instead of w^1_{ik} itself because $\sum_{i,k}(w^1_{ik})^2 = 1$ [11].

We define the weight S_k (k = 1, 2, …, N) to be the sum of all $(w^1_{ik})^2$ in the kth subvector $\boldsymbol{w^1_k}$, i.e.,

$$S_k = \sum_{i}^{L}(w^1_{ik})^2 . \qquad (4)$$

S_k represents the relative strength of the contributions of the flows towards the kth subnet. Thus, the knowledge of weight vector $\boldsymbol{S} = (S_1, S_2, \ldots, S_N)$ across varying k constitutes one summary view of network-wide traffic load. The evolving pattern of spatial-temporal correlation might allow us to infer where and when network congestion emerges.

3 Simulation Model

Network simulation plays a key role in building an understanding of network behavior. Choosing a proper level of abstraction for a model depends very much on the objective. Studying collective

phenomena seems to require simulating networks with a large spatial extent. Appropriate models for such studies should also include substantial detail representing protocol mechanisms across several layers of functionality (e.g., application, transport, network, and link), yet must also be restricted in space and time in order to prove computationally tractable. Previously, we adopted a two-tier modeling approach that maintains the individual identity of packets, producing a full-duplex "ripple effect" at the packet level, and that also accommodates spatial correlations in a regular network structure [12, 13]. While our two-tier model has been applied successfully to qualitatively understand some traffic characteristics in large-scale networks [12, 7], some doubts exist about the realism inherent in the regular network structure of such a model. In this paper, we retain the individual identity of packets but we replace the regular network structure of our previous two-tier model with a large-scale irregular topology chosen to resemble the topology of a real network.

3.1 Topology

Here, we transform our regular two-tier model into an irregular four-tier topology, as shown in Figure 1. (The host-computer tier is not shown in Figure 1.) While the network core becomes heterogeneous and hierarchical, (tier-four) host-computer behavior remains homogeneous at the edge of the network. The (tier-one) backbone topology, including eleven (backbone) routers (A, B, … K), resembles the original Abilene network, as described elsewhere [14]. Links between backbone routers have varying delays. For example, the longest link between backbone routers D and F has a 20-ms propagation delay; the shortest propagation delay (3 ms) exists on the link between backbone routers J and K. Forty (tier-two) subnets connect to the backbone through subnet routers, represented by designators such as A1 and B2. Each subnet contains a variable

number of (tier-three) leaf routers, such as A1a and B2b. Each leaf router supports an equal number (200 in this paper) of (tier-four) source hosts, and a variable number ($\leq$ 800 in this paper) of (tier-four) receivers, activated on demand. Link capacities gradually increase from host (tier four) to backbone, with (tier-one) backbone links being hundreds of times faster than links to (tier-four) hosts.

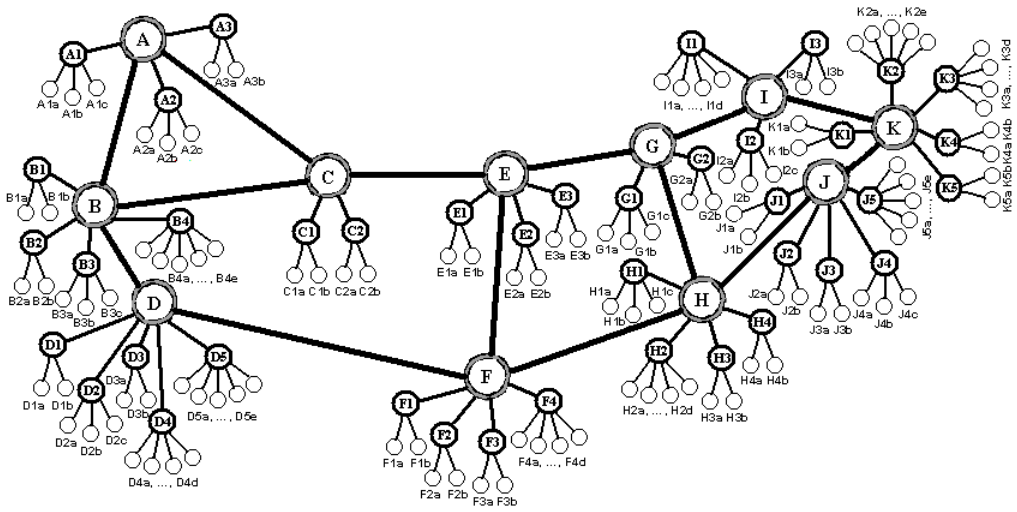

Figure 1: The simulation model with 11 backbone routers, 40 subnet routers, and 110 leaf routers

3.2 Traffic sources

There are a total of 22,000 sources in our model, which operates at the packet level. Each source models traffic generation as an ON/OFF process, which alternates between wake and sleep periods with average durations λ_{on} and λ_{off}, respectively. When awake, a source may send, subject to any restrictions imposed by TCP, one packet at each time-step to the source's attached leaf router. The packet will be placed at the end of the router's queue. At the beginning of each

ON period, a destination receiver is chosen randomly from among leaf routers other than the local routers, i.e., all flows must transit through at least one backbone link. When sleeping, the source does not generate new packets at each time-step. ON/OFF sources provide a convenient model of user behavior.

Empirical measurements on the Internet observe a heavy-tailed distribution of transferred file sizes [12]. Here, we use the Pareto distribution for both ON and OFF processes with the same shape parameter α [12]. In this paper, $\lambda_{on} = 50$, $\lambda_{off} = 5000$ and $\alpha = 1.5$.

3.3 Routers

Packets, the basic unit of transmission on TCP/IP networks, contain destination addresses, used by routers to correctly forward, and source addresses, used by receivers to identity the destination address for reply packets. To store and forward packets, which in our model travel a constant, shortest path between a source-destination pair for each flow, all routers maintain a queue of limited length (160 packets/router here), where arriving packets are stored until they can be processed: first-in, first-out. For convenience, in this paper we assume that every discrete simulation time-step is 1 millisecond. However, each leaf router, subnet router, or backbone router can in turn forward 5, 20, or 160 packets during one millisecond. This simulates capacity differences among various link classes from leaf-access to backbone in a hierarchically structured network. With such parameter settings, simulated backbone links are very lightly loaded.

We select several subnet routers as observation points, e.g., B4, D5, F4, I1, and J5, which record all outbound flows to destination leaf routers. In this paper, we assume that a central

collector reliably receives a continuous stream of measured data from observation points in time to perform analysis for our various experiments.

4 Simulation Experiments

To observe the macroscopic effect of DDoS attacks, we arrange 50 attack sources in our simulation model, which are distributed uniformly throughout the simulated network. In our experiments, there are a total of 22,000 source nodes, and more than 10,000 simultaneously active TCP connections; thus, DDoS flows cannot be easily identified from the legitimate background traffic.

Usually, DDoS attacks directed against the network infrastructure can lead to more widespread damage than those directed against individual web servers. Here, a subnet router (I1) will be the attack target. (Elsewhere [15], we report results where the attack target is a leaf router.) Since routers under attack may fail to collect and transfer measurement data, we assume in our experiments that the attack disables the observation point deployed at the subnet-router I1; thus, we perform our analysis using data from only four observation points (B4, D5, F4 and J5; $L = 4$). We experiment with various attack patterns: constant rate, increasing rate, natural-network-congestion-like, subgroup, pulsing, and TCP-targeted attacks. Figure 2 provides a pictorial representation of four attack classes. We describe each type of attack further in the appropriate sections, where we apply our analysis technique to these different attack scenarios.

4.1 Constant rate attack

Constant rate, the simplest attack technique, is typical of known DDoS attacks. We arrange for all the 50 attack sources to launch constant-rate attacks collectively (that is, simultaneously).

Here, we do not have the attack sources generate attack packets with full force [16], so that they cannot be easily identified through attack intensity at the source or in transit networks. We assume that the variable H represents the intensity of an attack source. Since sources can only create one packet every millisecond, the maximum attack rate is one packet per millisecond, i.e., $H \leq 1$ (packet/ms). Figure 2(a) shows the attack dynamics with a constant rate, starting from time t_0. We experiment with a constant-rate DDoS attack where H = 1/5, that is, each attack source creates one attack packet for every 5 milliseconds beginning from t_0 (= 500 s in this paper).

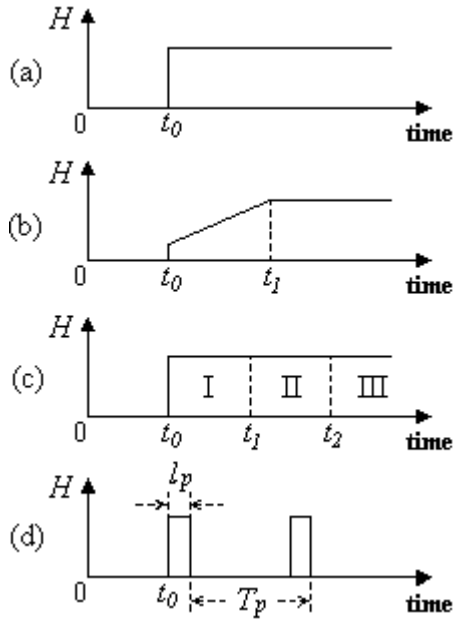

Figure 2: Attack dynamics for: (a) constant rate, (b) increasing rate, (c) subgroup, (d) pulsing

To show changes in spatial traffic pattern under this constant-rate attack (H = 1/5), we calculate the weight vector $\boldsymbol{S}$ using M data points within a moving time window MT from one time period to the next. Figure 3 shows $\boldsymbol{S}$ evolving with T = 2 s and with the time window MT (=

200 × 2 s = 400 s) sliding ahead every 20 s. The time axis indicates the end of the moving time window. The location axis represents 11 backbone-router zones, each of which denotes the subset of subnet routers therein. We can see that the attack really leads to a network-wide shift of spatial-temporal correlation, and the congestion at the victim (I1) reveals itself. Since we can observe this phenomenon and get the time and location of the attack (and without any help from the suffering victim), this type of monitoring should be meaningful to trigger further detection and filtering. On the other hand, Figure 3 also contrasts the distinct effect of the transient period (during onset of the attack) with the indistinctness after the new pattern becomes steady (say around t = 900 s). With fewer observation points, capturing the effect of transient periods is very important for monitoring the network-wide pattern shifting over time [7, 15].

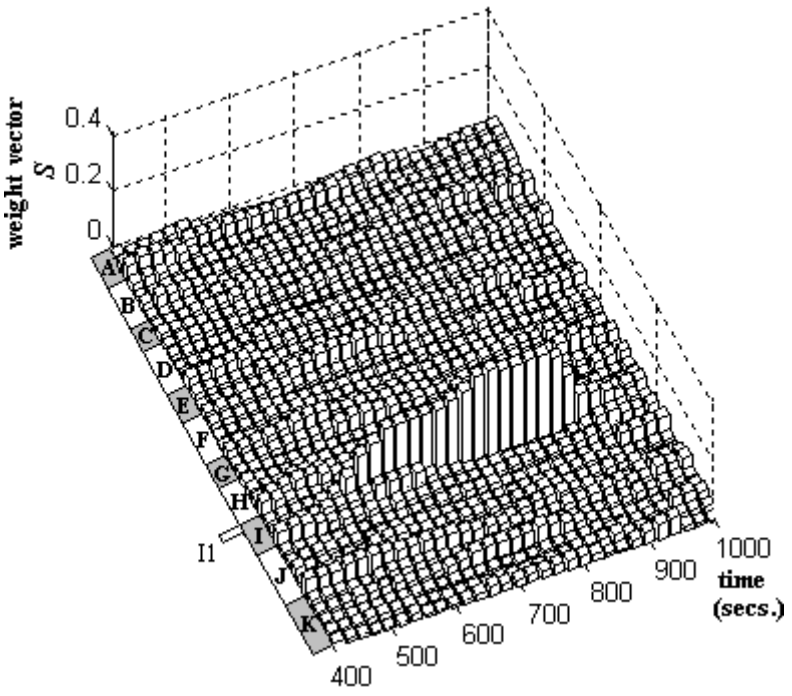

Figure 3: The spatial-temporal pattern of the constant-rate attack with H = 1/5

Moreover, the attack intensity H may influence the spatial-temporal dynamics. Figure 4 shows the spatial-temporal pattern of a constant-rate attack with H = 1/10, and the weaker visibility of I1 (compared against Figure 3, when H = 1/5). We can easily imagine an attack weak enough not to cause an apparent shift of spatial-temporal correlation.

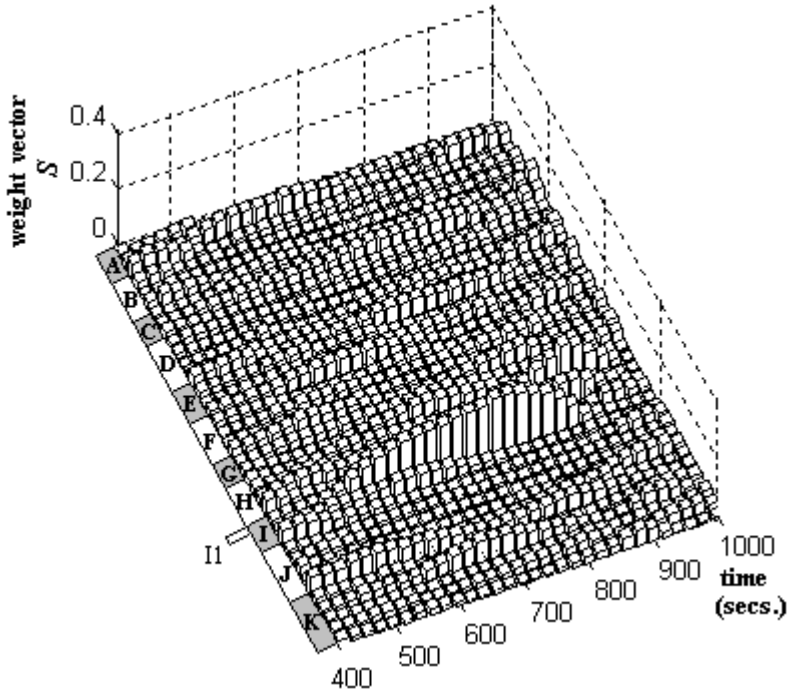

Figure 4: The spatial-temporal pattern of the constant-rate attack with H = 1/10

While current detection methods seem designed mainly to counter constant-rate DDoS attacks, attackers may choose more sophisticated attack dynamics. As defense mechanisms are deployed to counter simple attacks, we are likely to see more complex attack patterns that make countermeasures against DDoS attacks more difficult.

4.2 Increasing rate attack

Usually, an abrupt change in traffic volume is one important signal to initiate anomaly detection. Attacks that exhibit a gradually increasing rate, as shown in Figure 2(b), may lead to slow exhaustion of a victim's resources. The state change in the victim network could be so gradual that services degrade slowly over a long period of time, delaying detection of the attack. In the case of increasing-rate attacks, an abrupt change in traffic volume will not occur; thus, some other form of anomalous pattern must be identified.

In this experiment, we assume that the steady attack rate (H = 1/5) of each attack source is achieved gradually over 300 seconds (from t_0 = 500s to t_1 = 800s) starting from a weak rate (H = 1/35). Figure 5 shows the spatial-temporal pattern of this increasing-rate attack. The attack gradually builds up congestion in router I1 as attack intensity increases, and thus reveals itself in the spatial-temporal evolution. There might be a possibility that an increasing-rate attack deliberately accelerates slowly, taking a very long time to reach a steady intensity; however, we believe that the congestion caused by this class of attack will still be discovered sooner or later, along with the changing traffic demand [15].

4.3 Natural-network-congestion-like attack

Flash crowds on the Internet can trigger false alarms among DDoS attack detection mechanisms. Further, DDoS attacks may mimic natural network congestion in order to avoid detection [6]. Usually, legitimate traffic occurs in waves, while DDoS attacks, such as ICMP ping flooding, direct malicious traffic toward victims in a fairly persistent form. An experienced attacker may design each attack source to behave like a normal user, exhibiting bursts of traffic followed by silent periods, so that the flooding attack would appear similar to natural network congestion. In

particular, if attack packets use the same forged source addresses during each burst period, then the attacks cannot be detected simply by recognizing sudden change in the number of connections [17].

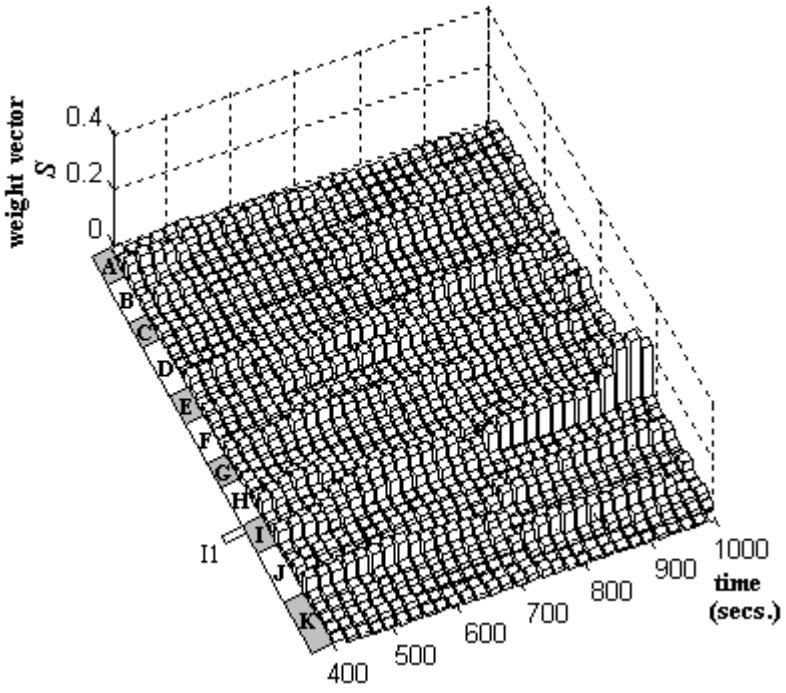

Figure 5: The spatial-temporal pattern of the increasing-rate attack with H = 1/5 increased from H = 1/35 over 300 seconds from t_0 = 500 s to t_1 = 800 s

Following the expression formulated by Huang, Kobayashi, and Liu [6], we can calculate the minimum number of attack sources required to deplete the link capacity (C_v packets/ms) of a victim,

$$N_s \geq C_v(\lambda'_{on} + \lambda'_{off}) / \lambda'_{on}, \qquad (5)$$

where λ'_{on} and λ'_{off} denote the mean ON and OFF periods for each source. In our experiments, the capacity of the victim subnet is 20 packets per millisecond, i.e., C_v = 20. If we set λ'_{on} = 5 ms and λ'_{off} = 50 ms, then $N_s \geq 220$. This means that 220 attack sources can completely block a victim's service by launching an attack that mimics natural network congestion. Here, we use only 50 such attack sources to degrade victim performance, and we observe if this low-grade attack leads to a corresponding shift in spatial-temporal pattern. Figure 6 shows the spatial-temporal pattern formed by this low-grade attack, which mimics natural network congestion. The spatial-temporal evolution reveals the induced congestion.

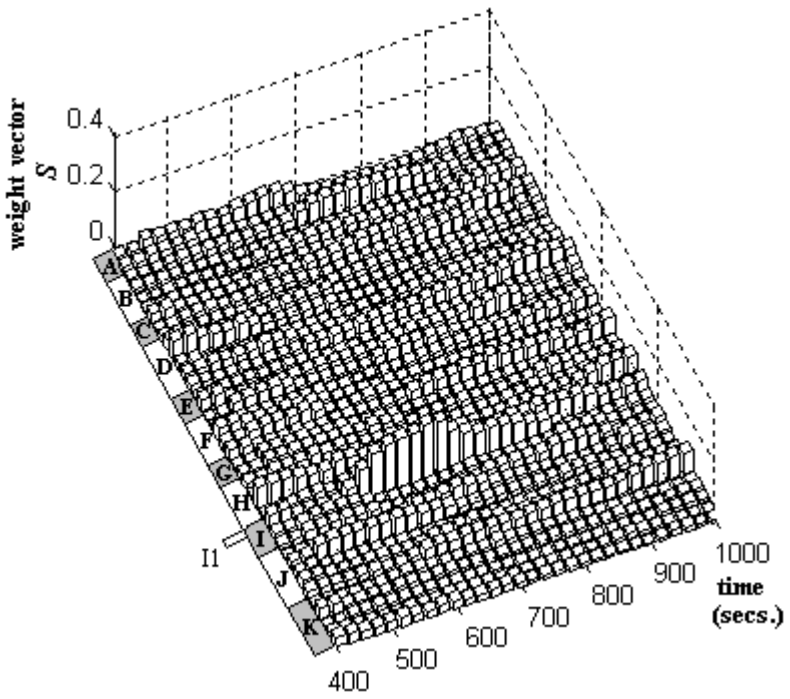

Figure 6: The spatial-temporal pattern of the natural-network-congestion-like attack with λ'_{on} = 5 ms, λ'_{off} = 50 ms, and N_s = 50

4.4 Subgroup attack

If compromised sources are divided into several subgroups that coordinate so that one subgroup is always active, then successive attacks by the subgroups can still induce continuous denial of service at an attack victim. To simulate a subgroup attack, we divide 50 attack sources into three subgroups. As shown in Figure 2(c), each of the three subgroups (I: 17 sources, II: 17 sources, and III: 16 sources) is active in turn: the first subgroup attacking from t_0 to t_1, the second subgroup attacking from t_1 to t_2, and the third subgroup starting from t_2. Here, we set t_0 = 500 s, t_1 = 680 s, t_2 = 860 s, and H = 1/5. We also arrange the three subgroups spatially in the left, the middle, and the right of the network so that the attack changes direction dynamically. Such attack dynamics make if difficult for *traceback* approaches [18, 19] to identify the attack sources and for *pushback* mechanisms [20] to capture the congestion signature.

Figure 7 shows the spatial-temporal pattern of our simulated subgroup attack, which reveals itself through the signature of congestion at victim I1. Comparing against the constant rate attack (Figure 3 and Figure 4), we find that for our analysis method the dynamic nature of the subgroup attack seems advantageous, because the increased correlation induced by shifts in attack traffic keeps the weight of the victim I1 salient over a longer time range.

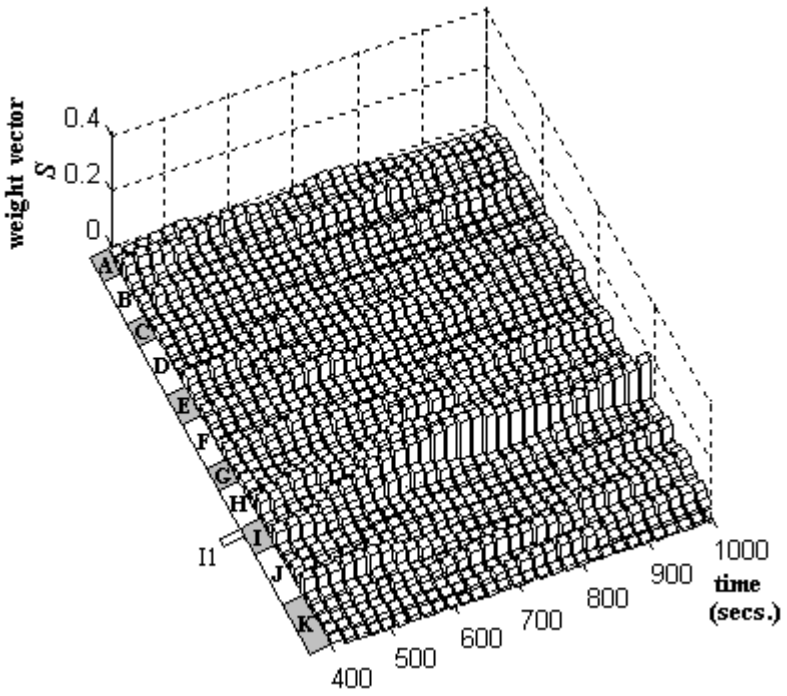

Figure 7: The spatial-temporal pattern of the subgroup attack with three subgroups and $H = 1/5$

4.5 Pulsing attack

Pulsing attacks, exhibiting a fluctuating rate oscillating between H and zero, occasionally reduce attack traffic in order to avoid detection. The dynamics of a pulsing attack appear as an ON/OFF pattern with period T_p and burst duration l_p, as shown in Figure 2(d). During a pulsing attack, attack sources periodically abort the attack only to resume it at a later time. Since sources of a pulsing attack send out bursts of attack packets instead of steady packet streams, the tradeoff between detection accuracy and latency impedes real-time detection.

In our simulated pulsing attacks, we set $T_p = 300$ s, $l_p = 60$ s, and $H = 1/5$. Figure 8 shows the resulting spatial-temporal pattern, where the attack is clearly revealed. Similar to the subgroup

attack, the dynamic nature of the pulsing attack leads to frequent shifts in the traffic pattern, which strengthens correlation and causes the greater weight of victim I1 to persist for a longer period.

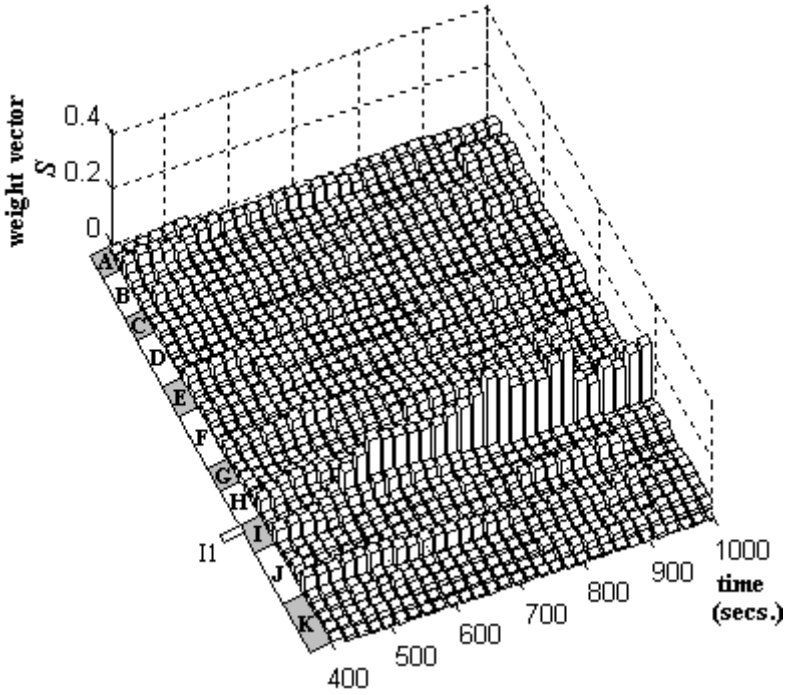

Figure 8: The spatial-temporal pattern of the pulsing attack with T_p = 300 s, l_p = 60 s, and H = 1/5

A sophisticated attacker may attempt to launch a special variant of the pulsing attack: a low-rate TCP-targeted DDoS attack [21], which exploits the TCP retransmission time-out mechanism to throttle TCP flows, while eluding detection. Under such an attack, TCP flows to the victim may continually incur loss as they try to exit their timeout states. The TCP-targeted DDoS attack transmits short-duration high-rate bursts periodically. We simulate such a TCP-targeted attack

with the same parameters used by Kuzmanovic and Knightly [21], i.e., T_p = 1.1 s, l_p = 100 ms, and H = 1. Figure 9, which shows the spatial-temporal pattern of this TCP-targeted DDoS attack, still reveals the induced congestion at the victim I1.

Note that the weight of the victim in Figure 9 does not benefit from the dynamics of this TCP-targeted attack. Figure 9 is similar to Figure 4 and Figure 6, which exhibit more natural patterns of congestion. In fact, the actual intensity ($\frac{l_p \times H}{T_p} = \frac{0.1 \times 1}{1.1} = \frac{1}{11}$) of our simulated TCP-targeted attack is the same as the intensity ($\frac{\lambda_{on}^{'}}{\lambda_{on}^{'} + \lambda_{off}^{'}} = \frac{5}{5+50} = \frac{1}{11}$) of our attack that mimics natural network congestion, which is similar to the intensity (H = 1/10) of one of our simulated constant-rate attacks (Figure 4). The TCP-targeted and the natural-congestion attacks both exhibit fluctuating rates but over small timescales (near or under 1 s), so the associated spatial-temporal traffic patterns tend to become steady more quickly, causing loss of the increased correlation associated with changing traffic patterns. However, the dynamics of the pulsing attack (Figure 8) varies over a larger time scale (about 300 s), so the increased correlation is present not only at onset of the attack, but continues to appear during the entire attack.

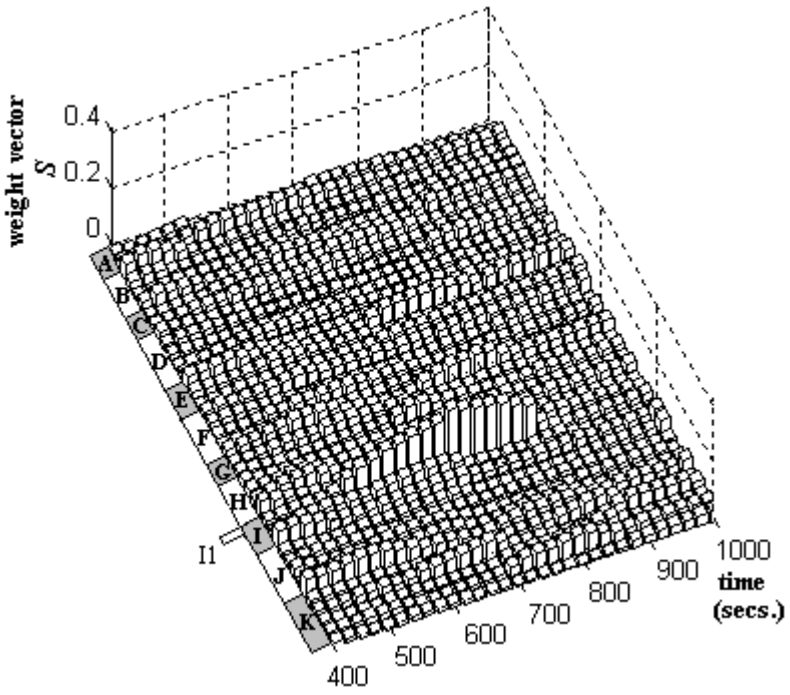

Figure 9: The spatial-temporal pattern of the TCP-targeted attack with T_p = 1.1s, l_p = 100 ms, and H = 1

5 Conclusion

Creating defenses for flooding-based DDoS attacks requires monitoring dynamic network activities in order to obtain timely and significant information. While much current effort focuses on detecting constant-rate attacks, DDoS attack patterns appear likely to become more sophisticated and effective. In this paper, we proposed a means for early detection of DDoS flooding attacks by monitoring macroscopic (network-wide) effects. We experimented with different attack modes: constant rate, increasing rate, natural-network-congestion-like, subgroup, pulsing, and TCP-targeted attacks. We found that these attacks, which have the apparent effect of

inducing network congestion, reveal themselves through shifts of spatial-temporal patterns that exhibit the same signature: congestion at the victim network. Our simulation results show that macroscopic-level monitoring could capture shifting traffic patterns during transient periods with only a few observation points. Our analysis method reveals the time and location of an attack without traffic observations from the suffering victim. We also find that the dynamic nature of selected attack types (e.g., subgroup and pulsing attacks) may be advantageous for our analysis method, because increased correlation induced by dynamic changes in attack traffic keeps the weight of the attack victim salient for longer periods. Our results suggest that macroscopic-level monitoring might be both practical and helpful for triggering more focused detection and filtering in transit or source networks.

Reference

[1] J. Saltzer, D. Reed, and D. Clark, End-to-end arguments in system design, ACM Trans. Computer System, 2(4), pp. 277-288, November 1984.

[2] D. Moore, G. Voelker, and S. Savage, Inferring Internet denial of service activity, In Proceedings of the USENIX Security Symposium, Washington, DC, USA, August 2001.

[3] J. Mirkovic, J. Martin and P. Reiher, A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms, UCLA CSD Technical Report, CSD-TR-020018, 2001.

[4] J. Mirkovic, G. Prier, and P. Reiher, Challenges of Source-End DDoS Defense, *Proceedings of NCA 2003*.

[5] A. Feldmann, A. C. Gilbert, W. Willinger and T. G. Kurtz, The changing nature of network traffic: Scaling phenomena, ACM SIGCOMM Computer Communication Review, Volume 28, No. 2, April 1998, pp. 5-29.

[6] Q. Huang, H. Kobayashi, and B. Liu, Analysis of a New Form of Distributed Denial of Service Attack, the 37th Annual Conference on Information Science and Systems (CISS'03), Johns Hopkins University, Baltimore, Maryland, Mar. 2003.

[7] J. Yuan, K. Mills, A cross-correlation based method for spatial-temporal traffic analysis, submitted, 2003.

[8] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2000.

[9] The MathWorks, Inc., Natick, MA, USA, MATLAB Ueser's Guide, 1998.

[10] M. Barthelemy, B. Gondran, and E. Guichard, Large scale cross-correlations in Internet traffic, Physical Review E 66 (2002) 056110.

[11] K. I. Goh, B. Kahng, and D. Kim, Spectra and eigenvectors of scale-free networks, Physical Review E 64 (2001) 051903.

[12] J. Yuan and K. Mills, Implication of Internet Traffic Characteristics for Network-Adaptive Distributed Systems, submitted, 2003.

[13] J. Yuan, K. Mills, Exploring Collective Dynamics in Communication Networks, Journal of Research of the National Institute of Standards and Technology 107 (2), 179-191, 2002.

[14] M. Crovella and E. Kolaczyk, Graph wavelets for spatial traffic analysis, in proceedings of IEEE Infocom 2003, San Francisco, CA, USA, April 2003.

[15] J. Yuan, K. Mills, Macroscopic Dynamics of Large-Scale Data Networks, in preparation.

[16] J. Mirkovic, G. Prier and P. Reiher, Attacking DDoS at the Source, Proceedings of ICNP 2002, pp. 312-321, Paris, France, November 2002.

[17] P. Barford and D. Plonka, Characteristics of network traffic flow anomalies, in Proceedings of ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA, USA, November 2001.

[18] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, Practical network support for IP traceback, In Proceedings of ACM SIGCOMM '2000.

[19] A. C. Shoeten, C. Partridge, L. A. Sanchez, C.e E. Jones, F. Tchakountio, S. T. Kent, and W. T. Strayer, Hash-based IP traceback, In Proceedings of the 2001 ACM SIGCOMM Conference, San Diego, California, U.S.A., August 2001.

[20] R. Mahajan, S. M. Bellovin, S. Floyd, J. Ioannidis, V. Paxson, and S. Shenker. Controlling high bandwidth aggregates in the network, Technical report, AT&T Center for Internet Research at ICSI (ACIRI) and AT&T Labs Research, February 2001.

[21] A. Kuzmanovic, E. W. Knightly, Low-Rate TCP-Targeted Denial of Service Attacks, In Proceedings of ACM SIGCOMM 2003, Karlsruhe, Germany, August 2003.