Ijesm
Consulting, help, relaxation

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES
&
MANAGEMENT

Detecting Flash Crowds Using Traffic Pattern

Sirish Kumar.M ,,PG student,QCET,Nellore,sirishmca@gmail.com
SYED AKHTAR M.Tech Assistant Professor , akhtar.sab@gmail.com

Abstract—Distributed Denial of Service (DDoS) attack is a critical threat to the Internet, and botnets are usually the engines behind them. Sophisticated botmasters attempt to disable detectors by mimicking the traffic patterns of flash crowds. This poses a critical challenge to those who defend against DDoS attacks. In our deep study of the size and organization of current botnets, we found that the current attack flows are usually more similar to each other compared to the flows of flash crowds. Based on this, we proposed a discrimination algorithm using the flow correlation coefficient as a similarity metric among suspicious flows. We formulated the problem, and presented theoretical proofs for the feasibility of the proposed discrimination method in theory. Our extensive experiments confirmed the theoretical analysis and demonstrated the effectiveness of the proposed method in practice.

Index Terms—DDoS attacks, flash crowds, similarity, discrimination.

1 Introduction

In this paper, we present a novel flow similarity-based approach to discriminate DDoS attacks from flash crowds, which remains an open problem to date. Distributed Denial of Service (DDoS) attacks pose a critical threat to the Internet. A recent survey [1] of the 70 largest Internet operators in the world demonstrated that DDoS attacks have increased dramatically in recent years. Moreover, individual attacks are becoming stronger and more sophisticated. Motivated by huge financial rewards, such as renting out their botnets for attacks or collecting sensitive information for malicious purposes, hackers are encouraged to organize botnets to commit these crimes [2]. Furthermore, in order to sustain their botnets, botmasters take advantage of various antiforensic techniques to disguise their traces, such as code obfuscation, memory encryption [3], fresh code pushing for resurrection [4], peer-to-peer implementation technology [5], [6], [7], or flash crowd mimicking [8], [9]. Flash crowds are unexpected, but legitimate, dramatic surges of access to a server, such as breaking news. One powerful strategy for attackers is to simulate the traffic patterns of flash crowds to fly under the radar. This is referred to as a *flash crowd attack*.

The work of discriminating DDoS attacks from flash crowds has been explored for around a decade. Previous work [8], [10], [11] focused on extracting DDoS attack features, and was followed by detecting and filtering DDoS attack packets by the known features. However, these methods cannot actively detect DDoS attacks. The current most popular defence against flash crowd attacks is the use of graphical puzzles to differentiate between humans and bots [12]. This method involves human responses and can be annoying to users. Xie and Yu tried to differentiate DDoS attacks from flash crowds at the application layer based on user browsing dynamics [13], [14]. Oikonomou and Mirkovic tried to differentiate the two by modeling human behavior [15]. These behavior-based discriminating methods work well at the application layer. However, we have not seen any detection method at the network layer, which can extend our defence diameter far from the potential victim.

There are a number of reports on the size and organization of botnets [5], [7], [16], [17]. Bots are caught by honeypots and analyzed thoroughly via inverse engineering techniques. Botnet infiltrations are further implemented to collect first-hand information about their activities [2], [18], and Wang et al. have even implemented a peer-to-peer-based botnet for research purposes [19].

We note the following facts concerning the current botnets after our thorough study:

1. The attack tools are prebuilt programs, which are usually the same for one botnet. A botmaster issues a command to all bots in his botnet to start one attack session. This can be evidenced from the literature of botnet [2], [4], [5], [17].
2. The attack flows that we observe at the victim end are an aggregation of many original attack flows, and the aggregated attack flows share a similar standard deviation as an original attack flow, and the flow standard deviation is usually smaller than

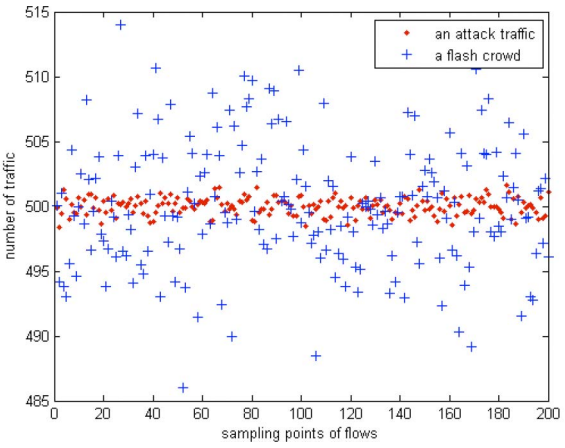

Fig. 1. The difference between an aggregated attack traffic and a flash crowd traffic under the current botnet size and organization.

that of genuine flash crowd flows. The reason for this phenomenon is that the number of live bots of a current botnet is far less than the number of concurrent legitimate users of a flash crowd. Rajab et al. recently reported that the live bots of a botnet is at the hundreds or a few thousands level for a given time point [20]. However, we observed that the number of concurrent users of the flash crowds of World Cup 98 is at the hundreds of thousands level (see the online supporting material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPDS.2011.262, for details) [21]. Therefore, in order to launch a flash crowd attack, a botmaster has to force his live bots to generate many more attack packets, e.g., web page requests, than that of a legitimate user. As a result, the aggregated attack flow possesses a small standard deviation compared with that of a flash crowd, which results in the phenomenon we see in Fig. 1.

Based on this observation, we found that the similarity among the current DDoS attack flows is higher than that of a flash crowd. Therefore, we propose a flash crowd attack detection method using the flow correlation coefficient. We aim to protect potential victims (e.g., web servers, mail servers) from flash crowd attacks within a community network. A community or ISP network often operates with the same Internet service provider domain or the virtual network of different entities which are all cooperating with one another. The community network benefits the defence of DDoS attacks in a wider range and in a cooperative way. This is hard to achieve in the realm of the Internet, where anarchy is the underlying principle. We first established a model for DDoS attack detection in a community network where the potential victim is situated. We then theoretically proved that attack flows can be discriminated from flash crowds under current botnet sizes and organization. Our experiments confirmed our theoretical conclusions.

The comparison among the proposed method and the previous ones can be found in the online supporting material.

This paper makes the following contributions:

- We found a new feature of flow similarity to defeat flash crowd attacks under current botnet size and organization. It is the first work in this field to the best of our knowledge. Within the relevant literature, flash crowd attacks continue to be a challenge. Our work sheds light on a new perspective in addressing this problem at the network layer.
- The proposed algorithm works independently of specific DDoS flooding attack genres. Therefore, it is effective against unknown forthcoming flooding attacks.
- The proposed correlation coefficient-based method is delay proof. This property is very effective against explicit random delay insertion among attack flows.
- We verified our observations with real data sets of flash crowds and real attack tool experiments in various scenarios. We conclude that it can effectively beat flash crowd attacks.

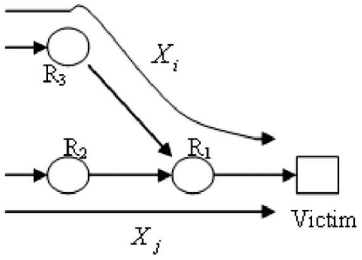

Fig. 2. A sample community network with network flows.

The remainder of the paper is organized as follows: the definitions and problem setting are presented in Section 2. The detection algorithm is proposed in Section 3. We analyze the proposed discrimination method in Section 4. Performance evaluations are conducted in Section 5. We discuss the possible antidetection methods in Section 6. Finally, we summarize the paper and present future work in Section 7.

2 Definitions and Problem Setting

In this section, we begin by presenting a number of preliminary definitions, and then discuss the setting of the discrimination problem.

For simplicity, we use the terms flow and network flow interchangeably in this paper.

Definition 1 (Network Flow). *For a given router in a local network (e.g., a community network), we cluster the network packets that share the same destination address as one network flow.*

A sample community network with flows can be found in Fig. 2. In the sample community network, R_2 and R_3 are the edge routers, and the server is the potential victim that we try to protect. There are two incoming flows, X_i and X_j observed at R_3 and R_2, respectively. They merge at router R_1 and both are addressed to the potential victim, and enter the community network via different paths. We sample the number of packets for a given network flow with a given time interval. Therefore, a network flow can be represented by a data sequence $X_i[n]$, where $i(i \geq 1)$ is the index of network flows, and n denotes the nth element in a data

sequence. For example, if the length of a given network flow X_i is N, then the network flow can be expressed as follows:

$$X_i = \{x_i[1], x_i[2], \ldots, x_i[N]\}, \quad (1)$$

where $x_i[k](1 \le k \le N)$ represents the number of packets that we counted in the kth time interval for the network flow. According to our definition of flow, a router may have many network flows at any given point in time.

Definition 2 (Flow Strength). *For a network flow X_i, let the length of the network flow be $N(N \ge 1)$. We define the expectation of the flow as the flow strength of X_i.*

$$E[X_i] = \frac{1}{N}\sum_{n=1}^{N} x_i[n]. \quad (2)$$

Flow strength represents the average packet rate of a network flow. If X_i is a DDoS attack flow, then we also call $E[X_i]$ *attack strength*.

Definition 3 (Flow Fingerprint). *For a given network flow X_i with length N, its fingerprint $X_i^{'}$ is the unified representation of X_i, namely,*

$$\begin{aligned} X_i^{'} &= \{x_i'[1], x_i'[2], \ldots, x_i'[N]\} \\ &= \left\{\frac{x_i[1]}{N \cdot E[X_i]}, \frac{x_i[2]}{N \cdot E[X_i]}, \ldots, \frac{x_i[N]}{N \cdot E[X_i]}\right\}. \end{aligned} \quad (3)$$

Following this definition, we know $\sum_{k=1}^{N} x_i^{'}[k] = 1$.

Based on Definitions 2 and 3, we obtain the following relationship between a network flow and its fingerprint

$$X_i = N \cdot E[X_i] \cdot X_i'. \quad (4)$$

As previously discussed, the current botnets, such as SDbot, Rbot and Spybot, employ the same program to generate attack packets. Furthermore, in order to achieve the purpose of denial of service, each bot has to generate as many attack packets as they can, usually with a very short delay (1 or 5 milliseconds) between two attack packets. This indicates that flow fingerprint does exist in attack flows for a given botnet.

Let X_i and X_j $(i \ne j)$ be two network flows with the same length N, then the correlation between the two flows is defined as

$$r_{X_i,X_j} = \frac{1}{N}\sum_{n=1}^{N} x_i[n]x_j[n]. \quad (5)$$

The correlation is used to describe the similarity of different flows. However, in some cases, it may indicate zero correlation although the two flows are completely correlated but with a phase difference. Therefore, the definition is modified to be practical as follows:

$$r_{X_i,X_j}[k] = \frac{1}{N}\sum_{n=1}^{N} x_i[n]x_j[n+k], \quad (6)$$

where $k(k = 0, 1, 2, \ldots, N-1)$ is the position shift of flow X_j.

However, there might still be a magnitude difference for the same similarity in different scenarios, therefore, unification is necessary.

Definition 4 (Flow Correlation Coefficient). *Let X_i and X_j $(i \ne j)$ be two network flows with the same length N. We define the correlation coefficient of the two flows as*

$$\rho_{X_i,X_j}[k] = \frac{r_{X_i,X_j}[k]}{\frac{1}{N}\left[\sum_{n=1}^{N-1} x_i^2[n] \sum_{n=1}^{N-1} x_j^2[n]\right]^{1/2}}. \quad (7)$$

The flow correlation coefficient is used to indicate similarity between two flows. It is sometimes the case that two similar flows may have a phase difference which will decrease the correlation coefficient. Fortunately, this is easy to deal with because we can shift one flow to match the other according to (6), and take the maximum value of the correlation coefficients to represent the similarity of two flows.

3 Similarity-Based Detection Method

In this section, we present the similarity-based detection method against flash crowd attacks.

For a given community network, we set up an overlay network on the routers that we have control over. We execute software on every router to count the number of packets for every flow and record this information for a short term at every router. Under this framework, the requirement of storage space is very limited and an online decision can be achieved.

A real community network may be much more complex with more routers and servers than the example network in Fig. 1. However, for a given server, we can always treat the related community network as a tree, which is rooted at the server. We must point out that the topology of the community network has no impact on our detection strategy, whether it is a graph or a tree, because our detection method is based on flows rather than network topology.

Once an access surge on the server occurs, our task is to identify whether it is a genuine flash crowd or a DDoS attack. According to our proposal, when a possible DDoS attack alarm goes off, the routers in the community network start to sample the suspected flows by counting the number of packets for a given time interval, for example, 100 milliseconds. When the length of a flow, N, is suitable, we start to calculate the flow correlation coefficient between suspected flows.

Suppose we have sampled M network flows, $X_1, X_2, \ldots, X_M$, therefore, we can obtain the flow correlation coefficient of any two network flows, $X_i(1 \le i \le M)$ and $X_j(1 \le j \le M, i \ne j)$. Let I_{X_i,X_j} be an indicator for the similarity of flow X_i and X_j, and I_{X_i,X_j} has only two possible values: 1 for DDoS attacks and 0 otherwise. Let δ be the threshold for the discrimination, then we have

$$I_{X_i,X_j} = \begin{cases} 1, & \rho_{X_i,X_j}[k] \ge \delta, \\ 0, & otherwise, \end{cases} \quad (8)$$

where $1 \le i, j \le M$, and $i \ne j$.

In general, we may have more than two suspected flows in a community network. This means we can conduct a number of different pairwise comparisons, and the final decision can be derived from them in order to improve the

reliability of our decision. We can, therefore, have an integrated DDoS attack positive probability as follows:

$$Pr(I_A = 1) = \frac{\sum_{1 \le i,j \le M, i \ne j} I_{X_i,X_j}}{\binom{M}{2}}, \quad (9)$$

where I_A is the indicator for DDoS attacks, and $I_A = 1$ represents positive for DDoS attacks. We can set a threshold $\delta^{'}(0 \le \delta^{'} \le 1)$ for our global judgement, therefore, we make our final decision with global information as follows:

$$I_A = \begin{cases} 1, & Pr(I_A = 1) \ge \delta^{'}, \\ 0, & Pr(I_A = 1) < \delta^{'}. \end{cases} \quad (10)$$

The value of $\delta^{'}$ has an impact on our detection accuracy. For example, if $\delta^{'} = 0.6$, then it is a DDoS attack if at least 60 percent of the comparisons are positive.

The detail of the detection algorithm can be found in the online supporting material.

4 Analysis on the Proposed Method

In this section, we first prove that flash crowds and DDoS attacks can be differentiated using the flow correlation coefficient in theory. Following this foundation, we analyze the effectiveness of the proposed discrimination method, and prove that the threshold δ in (8) does exist.

In order to make our analysis clear, we make the following assumptions:

1. There is only one server in a community network which is under attack or experiencing a flash crowd at any given time.
2. The attack packets enter the community network via a minimum of two different edge routers.
3. In one attack session, all the attack packets are generated by only one botnet, therefore the fingerprints of the attack flows are the same.
4. The network delays are discrete and countable.

Based on our knowledge of current botnets, the above assumptions are applicable in practice. However, attackers may disable our detection method by circumventing some conditions (e.g., the size of live bots) once our strategy is known to them. We will discuss this further in Section 6.

Theorem 1. *Let X_i and $X_j(i \ne j)$ be two traffic flows that share the same distribution, and the standard deviation σ is a random variable, the correlation coefficient of the two flows is inversely proportional to σ, namely, $\rho_{X_i,X_j} \propto \frac{1}{\sigma}$.*

The proof of Theorem 1 can be found in the online supporting material.

Based on Theorem 1, it is certain that we can differentiate DDoS attack flows from flash crowds as the standard deviation between these two phenomenons are different.

We now investigate the flow correlation coefficient of any two independent network flows, such as flash crowd flows. Previous research has demonstrated that web traffic follows the Pareto law [22], [23], hence, the Pareto distribution represents the flow fingerprint of flash crowds. The definition of the distribution is as follows.

Let X be a random variable, and x_m be the minimum time interval for arrival packets. For a given time interval x, the probability density function of the Pareto distribution is defined as

$$Pr[X = x] = \alpha \cdot x_m^{\alpha} \cdot x^{-(\alpha+1)}, \quad (11)$$

where $x_m \le x$, and α is the *Pareto index*.

Theorem 2. *Given two same length instances, X_i and $X_j(i \ne j)$, of a flash crowd that are generated by the same function and same parameters, $\lim_{N\to\infty} \rho_{X_i,X_j}[k] = 0$.*

The proof of Theorem 2 can be found in the online supporting material.

Theorem 2 shows that for any two independent flash crowds flows with length N, the flow correlation coefficient approaches 0 when N goes infinity.

We can easily obtain the following corollary by extending Theorem 2.

Corollary 1. *For two independent flash crowds X_i and X_j with the same length N, $\forall\delta(\delta < 1)$, $\exists N'$, when $N > N'$, we have $\rho_{X_i,X_j}[k] < \delta$.*

We now move to explore the flow correlation coefficient among DDoS attack flows. Let us first find the expression of a DDoS attack flow, X_i, which we obtained at an edge router. Suppose the observed attack flow is a mixture of attack flows that came from K different bots, and let $X_0^{'}$ be the fingerprint of the attack flows. Based on the aforementioned discussion, the fingerprint of different attack flows in one attack session is the same, except that there are delays in different attack flows. Let $X_0^{'}[j]$ represent the fingerprint that is delayed by j time units. As a result, the observed attack flow can be denoted as follows:

$$\begin{aligned} X_i &= \sum_{j=0}^{K} N \cdot E[X_i] \cdot X_0^{'}[j] \\ &= \sum_{j=0}^{k^{'}} a_j \cdot X_0^{'}[j], \end{aligned} \quad (12)$$

where $a_j(1 \le j \le k^{'} \le K)$ represents the magnitude of the attack flows that possess the same delay j at the edge router.

Theorem 3. *Let X_0' be the fingerprint of attack flows for one attack session. Under the condition of no network delay and no background noise, for two mixed attack flows X_i and $X_j(i \ne j)$ that we observed at two edge routers, the correlation coefficient of X_i and X_j is 1, namely, $\rho_{X_i,X_j}[k] = 1$.*

The proof of Theorem 3 can be found in the online supporting material.

Theorem 3 demonstrates that in the ideal conditions of a delay and noise free environment, any two DDoS attack flows from one botnet are totally correlated because they are a combination of attack flows from different bots with different network routes.

In reality, however, delay and noise do exist and bots in a centralized botnet are coordinated by their botmaster. This means the delays among the attack flows from different bots depend on normal Internet delays, and therefore are limited compared to fast Internet transportation facilities.

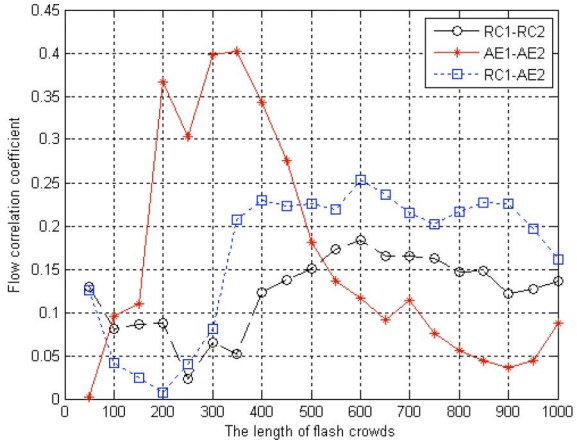

Fig. 3. The flow correlation coefficient against length of flows in the World CUP 98 data set.

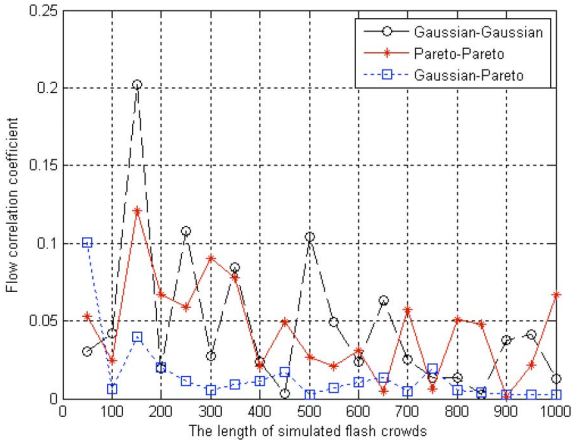

Fig. 4. The flow correlation coefficient against length of flows in general cases.

As a result, the delay free condition can be satisfied to some degree. On the other hand, noise in attack flows are the legitimate packets that are also addressed to the victim at the same time when a DDoS attack is ongoing. However, the strength of noise is much smaller compared with that of DDoS attack flows.

Following Theorem 3, we further have the following corollary.

Corollary 2. *Let Y_i and Y_j be the noises for two DDoS attack flows X_i and X_j of one attack session, $\forall\delta(\delta < 1)$, $\exists\Delta$, $\rho_{X_i,X_j}[k] \geq \delta$ holds when $\frac{E[X_i]}{E[Y_i]} > \Delta$ and $\frac{E[X_j]}{E[Y_j]} > \Delta$.*

The proof of Corollary 2 can be found in the online supporting material.

Corollary 2 indicates that the correlation coefficient of DDoS attack flows approaches 1 if the Signal-Noise-Ratio (SNR), $\frac{E[X_i]}{E[Y_i]}$, is sufficiently large. It is true that $E[X_i] \gg E[Y_i]$ and $E[X_j] \gg E[Y_j]$ for the DDoS attack cases, therefore, the correlation coefficient of attack flows is close to 1 in an ongoing DDoS attack scenario.

Theorem 4. *DDoS attack flow can be discriminated from flash crowds by the flow correlation coefficient at edge routers under two conditions: the length of the sampled flow is sufficiently large, and the DDoS attack strength is sufficiently strong.*

The proof of Theorem 4 can be found in the online supporting material.

It is necessary that we obtain an upper bound, δ, of the flow correlation coefficient for flash crowds for a given flow length. In the case that the flow correlation coefficient is greater than δ, we assume them to be DDoS attack flows.

5 Performance Evaluation

In this section, we demonstrate the effectiveness of the proposed detection method. We investigate the issue with a real data set first, followed by more general studies in order to achieve general results.

We used the 1998 FIFA World CUP data set [21] as a representative of flash crowds, which was collected at the official web server as one flow. The World Cup data set is a highly reliable flash crowd as DDoS attacks emerged in the year 2000. It has also been widely used for recent high-quality publications, such as [13], [14].

The flash crowd phenomenon can be found in the online supporting material.

We employed Mstream [24], a real DDoS attack tool, to generate the DDoS attack data set in an isolated network. For each data set, we counted the number of packets which were addressed to either the server of flash crowds or the victim of DDoS attacks.

We first examined two flash crowds of two knockout games in the World Cup 98 data set, Romania versus Croatia (RC in short) and Argentina versus England (AE in short). These two games were separated by 2 hours with both causing flash crowds. We took two segments from each game around the peak of the flash crowds with each segment lasting 1,000 seconds. We compared the flow correlation coefficient for two flows in various ways: two flash crowds for the same game—RC1 versus RC2 and AE1 versus AE2; two flash crowds from different games, RC1 versus AE2. The results are shown in Fig. 3. However, the data sets that we investigated were only two instances of many possible flash crowds. In order to have a general understanding of the variation of flow correlation coefficient against length of flows, we performed simulations in general cases: we examined the subject with two Gaussian flows, two Pareto flows, one Gaussian flow, and one Pareto flow with different parameters, respectively. The reason that we chose these two distributions for the simulation is that the Pareto distribution has been identified by researchers as the best one to represent network traffic. The Gaussian distribution is a general distribution in nature, and combinations of Gaussian distributions with different parameters can approximate other distributions. The results of the simulations are shown in Fig. 4. We also found that the flow correlation coefficient of two flows from the same distribution law (e.g., two Pareto flows with different parameters) is usually higher than that of two flows from different distribution laws (e.g., one from the Pareto distribution and another one from the Gaussian distribution). This indicates that the flow correlation coefficient decreases if the attack flows come from different botnets. From Figs. 3 and 4, we found that in general the flow correlation coefficient decreases when the length of flows

ISSN:2277-5528

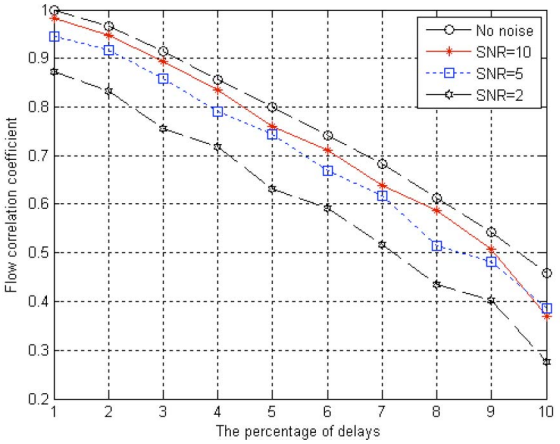

Fig. 5. The flow correlation coefficient of attack flows against background noise and delays.

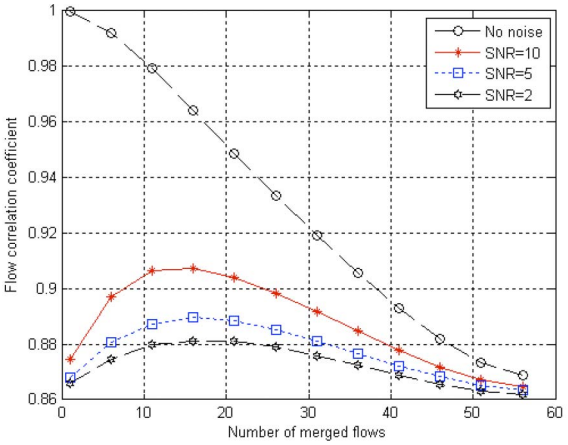

Fig. 6. The flow correlation coefficient of attack flows against a number of merged attacks and background noise.

increases, which confirms the results of Theorem 2. The real data experiments and the general simulations also indicated that the correlation coefficient of two flash crowds is less than 0.3 when the length of two flash crowds is greater or equal to 500. In other words, we can use $\delta = 0.3$ as the upper bound in practice for two flash crowds when the length is 500 or greater.

We now turn our attention to examining the flow correlation coefficient of DDoS attack flows. We setup two groups of DDoS attack machines using Mstream [24] as an attack tool in an isolated network, and collected two attack flows from the victim. We counted the number of packets in attack flows for every 100 milliseconds and collected 600 samples for each flow. In other words, our experiments lasted 60 seconds, and we explicitly controlled the delays between the two attack flows from 1 up to 10 percent (0.6 to 6 seconds). Furthermore, we also examined the impact on the flow correction coefficient of the two attack flows from background noise traffic. We used signal-noise-ratio to represent the ratio of attack flow strength and noise flow strength. In this experiment, the SNR is set as ∞ (no noise), 10, 5, and 2, respectively. The results are shown in Fig. 5.

From Fig. 5, we can confirm the conclusion of Theorem 2. The flow correlation coefficient is 1 if there is no noise and no delay between two attack flows. Background traffic noise contributes to the decrease of the flow correlation coefficient. According to [25], the strength of DDoS flooding attacks is usually more than 10 times the strength of normal legitimate flows (the noise in our experiment), meaning the condition of $SNR \geq 10$ is usually met for DDoS flooding attacks. Fig. 5 indicates that even when $SNR \geq 10$, the drop of the correlation coefficient is very limited (less than 0.05) compared to noise free cases.

We have to point out that although Fig. 5 shows that the flow correlation coefficient drops significantly against delay, it is not a problem for the proposed discrimination method. This is because we can shift one attack flow to calculate the flow correlation coefficient which is an advantage of the proposed method.

Another issue of DDoS attacks is the merging of the original attack flows on their way to the victims. We know that two original DDoS attack flows from one attack session share the same fingerprint, and they are 100 percent correlated if there is no noise. However, the attack flows that we observe at an edge router may be a mixture of a number of original attack flows with different delays. In this case, the correlation coefficient of these mixed flows will drop. In order to investigate this, we explicitly sampled 60 different attack flows with 1 time unit delay occurring one after the other. Therefore, the flow with the maximum delay was sampled 6 seconds later as the attack had been launched. We measured the variation of the flow correlation coefficient against the number of merged attack flows. We examined the subject under different conditions of background network traffic (noise) as we did in the previous experiment. We injected background traffic into every merged flow with different SNR, respectively. The results are shown in Fig. 6. This indicates that mergence of attack flows has a limited impact on the flow correlation coefficient even when there is strong noise. For example, the flow correlation coefficient is greater than 0.86 even when there is a converging of up to 60 different delayed flows with $\mathrm{SNR} = 2$. Combining our experiments and the analysis in the previous section, we are able to make the following conclusions: DDoS attacks can be discriminated from genuine flash crowds using the proposed flow similarity-based algorithm under the current botnet size and organization.

6 Discussion on Antidetections

As we know, detection and antidetection is an endless battle between defenders and attackers. Our discrimination method is effective under the current conditions of botnet size and organization. Hackers may make efforts to circumvent our similarity-based detection. We discuss them here for readers to carry on further research in this field.

First of all, if attackers are able to organize a super botnet, in which the number of live bots is the same or close to the number of concurrent users of a flash crowd, then, one bot can mimic the legitimate behavior of one user. As a result, the phenomenon that we saw in Fig. 1 does not exist any more, and the similarity among attack flows should be the same or very close to that of flash crowd flows.

We have to note that it is still an open problem for both attackers and defenders: Can botnet owners organize this kind of super botnet or not? There are many factors that

limit the number of live bots of a botnet, such as time zone, antivirus software, operating system patching.

Second, in order to disguise their flow fingerprints, bot writers may include many attack packet generation functions in their binary, and make each bot randomly choose one function to generate the attack packets. As we have seen in Fig. 5, flow similarity drops among different distribution flows compared with that of the same distribution flows. However, this impact is limited compared with that from the number of live bots.

Moreover, we believe there must be some differences between a mimicking attack and a genuine flash crowd. What we need to do is to discover them and deploy them to defeat mimicking attacks.

7 Summary and Future Work

In this paper, we tried to discriminate flash crowd attacks from genuine flash crowds, which is a tough and open problem for researchers. We found that DDoS attack flows possess higher similarity compared with that of flash crowd flows under the current conditions of botnet size and organization. We used the flow correlation coefficient as a metric to measure the similarity among suspicious flows to differentiate DDoS attacks from genuine flash crowds. We theoretically proved the feasibility of the proposed detection method, and our experiments confirmed the effectiveness of the discrimination method within the current botnet size and organization. We also discussed the possible antidetection methods from the attackers' perspective.

In regards to future work, we are very interested in working on the following issues. First of all, we are keen to investigate the possibility of organizing a super botnet, which has a sufficiently large number of live bots to beat the proposed method. Second, the tradeoff between detection accuracy and cost deserves a further investigation. Third, once our detection strategy is known to attackers, they may develop new strategies to disable our detection. It is necessary to explore which actions should we take against attackers' actions.

References

[1] Arbor, "IP Flow-Based Technology," http://www.arbornetworks.com, 2011.

[2] B. Stone-Gross, M. Cova, L. Cavallaro, B. Gilbert, M. Szydlowski, R. Kemmerer, C. Kruegel, and G. Vigna, "Your Botnet Is My Botnet: Analysis of a Botnet Takeover," *Proc. ACM Conf. Computer Comm. Security,* 2009.

[3] N. Ianelli and A. Hackworth, "Botnets as Vehicle for Online Crime," *Proc. 18th Ann. First Conf.,* 2006.

[4] C.Y. Cho, J. Caballero, C. Grier, V. Paxson, and D. Song, "Insights from the Inside: A View of Botnet Management from Infiltration," *Proc. Third USENIX Conf. Large-Scale Exploits and Emergent Threats: Botnets, Spyware, Worms, and More (USENIX LEET),* 2010.

[5] V.L.L. Thing, M. Sloman, and N. Dulay, "A Survey of Bots Used for Distributed Denial of Service Attacks," *Proc. SEC*, pp. 229-240, 2007.

[6] T. Holz, M. Steiner, F. Dahl, E. Biersack, and F.C. Freiling, "Measurements and Mitigation of Peer-to-Peer-Based Botnets: A Case Study on Storm Worm," *Proc. First Usenix Workshop Large-Scale Exploits and Emergent Threats (LEET),* 2008.

[7] M. Bailey, E. Cooke, F. Jahanian, Y. Xu, and M. Karir, "A Survey of Botnet Technology and Defenses," *Proc. Cybersecurity Applications and Technology Conf. for Homeland Security,* 2009.

[8] J. Jung, B. Krishnamurthy, and M. Rabinovich, "Flash Crowds and Denial of Service Attacks: Characterization and Implications for CDNs and Web Sites," *Proc. 11th Int'l Conf. World Wide Web (WWW),* pp. 252-262, 2002.

[9] A. Scherrer, N. Larrieu, P. Owezarski, P. Borgnat, and P. Abry, "Non-Gaussian and Long Memory Statistical Characterizations for Internet Traffic with Anomalies," *IEEE Trans. Dependable Secure Computing,* vol. 4, no. 1, pp. 56-70, Jan.-Mar. 2007.

[10] G. Carl, G. Kesidis, R. Brooks, and S. Rai, "Denial-of-Service Attack-Detection Techniques," *IEEE Internet Computing,* vol. 10, no. 1, pp. 82-89, Jan./Feb. 2006.

[11] Y. Chen and K. Hwang, "Collaborative Detection and Filtering of Shrew DDoS Attacks Using Spectral Analysis," *J. Parallel Distributed Computing,* vol. 66, no. 9, pp. 1137-1151, 2006.

[12] S. Kandula, D. Katabi, M. Jacob, and A. Berger, "Botz-4-Sale: Surviving Organized DDoS Attacks that Mimic Flash Crowds (Awarded Best Student Paper)," *Proc. Second Symp. Networked Systems Design and Implementation (NSDI '05),* 2005.

[13] Y. Xie and S.-Z. Yu, "A Large-Scale Hidden Semi-Markov Model for Anomaly Detection on User Browsing Behaviors," *IEEE/ACM Trans. Networking,* vol. 17, no. 1, pp. 54-65, Feb. 2009.

[14] Y. Xie and S.-Z. Yu, "Monitoring the Application-Layer DDoS Attacks for Popular Websites," *IEEE/ACM Trans. Networking,* vol. 17, no. 1, pp. 15-25, Feb. 2009.

[15] G. Oikonomou and J. Mirkovic, "Modeling Human Behavior for Defense against Flash-Crowd Attacks," *Proc. IEEE Int'l Conf. Comm.,* 2009.

[16] C. Patrikakis, M. Masikos, and O. Zouraraki, "Distributed Denial of Service Attacks," *The Internet Protocol J.,* vol. 7, no. 4, pp. 13-35, 2004.

[17] T. Peng, C. Leckie, and K. Ramamohanarao, "Survey of Network-Based Defense Mechanisms Countering the DoS and DDoS Problems," *ACM Computing Survey,* vol. 39, no. 1, pp. 123-128, 2007.

[18] D. Dagon, C. Zou, and W. Lee, "Modeling Botnet Propagation Using Time Zones," *Proc. 13th Network and Distributed System Security Symp. (NDSS),* 2006.

[19] P. Wang, S. Sparks, and C.C. Zou, "An Advanced Hybrid Peer-to-Peer Botnet," *IEEE Trans. Dependable and Secure Computing,* vol. 7, no. 2, pp. 113-127, Apr.-June 2010.

[20] M.A. Rajab, J. Zarfoss, F. Monrose, and A. Terzis, "My Botnet is Bigger than Yours (Maybe, Better than Yours): Why Size Estimates Remain Challenging," *Proc. First Conf. First Workshop Hot Topics in Understanding Botnets (HotBots '07),* 2007.

[21] WorldCup98, http://ita.ee.lbl.gov/html/contrib/WorldCup.html. 2011.

[22] V. Paxson and S. Floyd, "Wide Area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Trans. Networking,* vol. 3, no. 3, pp. 226-244, June 1995.

[23] M.E. Crovella and A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes," *IEEE/ACM Trans. Networking,* vol. 5, no. 6, pp. 835-846, Dec. 1997.

[24] G. Cheng, "Malware FAQ: Analysis on DDoS Tool Stacheldraht v1.666," http://www.sans.org/resources/malwarefaq/Stacheldraht.php. 2011.

[25] D. Moore, C. Shannon, D.J. Brown, G.M. Voelker, and S. Savage, "Inferring Internet Denial-of-Service Activity," *ACM Trans. Computer Systems,* vol. 24, no. 2, pp. 115-139, 2006.