An Adversary-Centric Behavior Modeling of DDoS Attacks

An Wang
George Mason University
awang10@gmu.edu

Aziz Mohaisen
University at Buffalo
amohaisen@gmail.com

Songqing Chen
George Mason University
sqchen@gmu.edu

***Abstract*—Distributed Denial of Service (DDoS) attacks are some of the most persistent threats on the Internet today. The evolution of DDoS attacks calls for an in-depth analysis of those attacks. A better understanding of the attackers' behavior can provide insights to unveil patterns and strategies utilized by attackers. The prior art on the attackers' behavior analysis often falls in two aspects: it assumes that adversaries are static, and makes certain simplifying assumptions on their behavior, which often are not supported by real attack data.**

In this paper, we take a data-driven approach to designing and validating three DDoS attack models from temporal (e.g., attack magnitudes), spatial (e.g., attacker origin), and spatiotemporal (e.g., attack inter-launching time) perspectives. We design these models based on the analysis of traces consisting of more than 50,000 *verified* DDoS attacks from industrial mitigation operations. Each model is also validated by testing its effectiveness in accurately predicting future DDoS attacks. Comparisons against simple intuitive models further show that our models can more accurately capture the essential features of DDoS attacks.

I. Introduction

Distributed Denial of Services (DDoS) attacks present one of the most persistent and damaging threats on the Internet, despite many research and developed efforts towards analysis, characterization and defenses in the past decade [1], [2], [3], [4]. It is widely believed that connectivity and bandwidth are the highest single cost item in today's enterprises operations [5]. A large contributor to this cost are DDoS attacks, which hinder the availability of Internet services, and increase their cost of operation [6]. Such targets are not limited to a certain group of enterprises, and today cover all verticals of businesses, including banks, energy and utility companies, web and network service providers, among others. Unlike other malicious activities on the Internet that are perhaps simpler to understand [7], [8], [9], [10], DDoS attacks are more complex phenomenon, highlighted in multiple ways: varieties of malware utilized by botnet families for launching attacks [11], [12], [13], the source and structure of those botnets [14], [15], [16], and the attack traffic mechanisms utilized to launch the attacks [17], [18], [19].

Botnets used for launching attacks are typically distributed or peer-to-peer systems, making traditional takedown strategies less effective [20]. Moreover, typical DDoS attacks today are not isolated acts, but different botnet families may collaborate with each other, highlighting a more sophisticated ecosystem [21], [22], [23]. As a result, and despite the wide spectrum of efforts pursued to defend against DDoS attacks [17], [2], [24], the complexity and the sophistication of the ecosystem of malicious actors launching those attacks are a fundamental cause of their persistence today. Even worse, such defenses prove less effective, and the defense posture deteriorates especially when targeted services only have limited access to threat intelligence (based on actual historical attack data). Eventually, targets still count on reputation or whitelist-based systems for protection [25], [26], [27].

For better defenses, DDoS attacks and their evolution need to be understood. On the one hand, botnet-based attacks are sophisticated, and strategies used by botmasters keep evolving [21]. On the other hand, DDoS attacks launched by those botnets can be viewed as output processes of such (distributed) systems (botnets) based on relevant input variables. However, understanding DDoS attacks through static snapshots analysis and incident reports [28], [29], [30], while meaningful, does not address the long term questions for a better defense posture. This line of thoughts motivated several prior research efforts for modeling the behavior of DDoS attacks (see §VIII), although such efforts often fall short in: 1) they consider DDoS attacks in simplistic scenarios, and 2) they consider DDoS attacks and attackers' behavior constant or static over time, which is often contradicted with reality. As a result, modeling DDoS attack behaviors has been a difficult task because of two reasons 1) dimensionality of the feature space used for modeling attacks (which is often high), and 2) the dynamic nature of DDoS attacks, which is not captured in prior work.

DDoS attacks are complex by nature, and the state-of-the-art makes certain simplifications that may deviate from reality, making the resulting models less practical. For example, Du *et al.* [31] and Qin *et al.* [32] considered DDoS attacks as fingerprints in a sequence of network events, and linear correlations between multiple attacks were extracted for attack inferences. However, DDoS attacks are complex; they are featured by multidimensional space of both temporal and spatial features that are not captured by this simplification. For instance, the current DDoS attack may correlate with both previous DDoS attacks towards the same target and other attacks in the same network neighborhood. Such correlations are not always linear either. Thus, current implementations fail to uncover such aspects in DDoS attacks for realistic modeling.

A large body of the prior art uses game theoretic-based models, including Zang *et al.*'s [33] and Yan *et al.*'s [34], which assume that resource optimization applies to strategic decision making for attackers. However, sophisticated attackers can employ evasive strategies to avoid detection and mitigation. For example, attack durations may depend on the botnet family utilized as well as the bot activities, both of which could change dynamically over time. Also, DDoS attackers use polymorphism, which is hard to capture by optimization models. Without actual datasets, all of those dynamic features are undermined.

This work. In this paper, we tackle those limitations. Based on more than 50,000 *verified* DDoS attack traces from industrial mitigation operations, we propose three data-driven models that can capture the temporal, spatial and spatiotemporal behaviors of DDoS attacks characterized by their dynamics, and ultimately help with insight into defenses and attack remedies. For this purpose, we reverse-engineer DDoS attacks by in-depth analysis across multiple principal features that are associated with the attackers and targets. For attackers, botnet activities and active bots are the major contributors to the model. For the target, target-related features, such as the network neighborhood and type of service, determine attack durations as well as attack mechanisms employed on the target by the attacker. Furthermore, location features have greater impact on the botnet families utilized in DDoS attacks [21].

Contributions. Our contributions in this paper include the following. 1) We design and validate temporal, spatial, and spatiotemporal models that can accurately capture DDoS dynamics. 2) Our proposed models can be used to predict essential features of future DDoS attacks. Thus, it can greatly help the defense of relevant stakeholders. 3) Our models alleviate the limited information available for a monitoring and defense entity by revealing various correlations between modeled variables and the attack features of interest. Such an approach of understanding DDoS attacks with limited information becomes appealing with the evolution of cloud-based security services, which do not have access to contexts nor complete data concerning attacks [3].

Organization. An overview of the dataset used in this work and the collection method are outlined in §II. Modeling is initiated in §III: new characteristics that capture DDoS attacks by botnets are presented through an in-depth analysis of actual DDoS traces §III-A and associated variables outlined formally and quantified for modeling attacks in §III-B We use the previously defined variables to build analytical models for predicting certain aspects of those attacks, including temporal models §IV, spatial model §V, and spatiotemporal models §VI. Discussion is introduced in §VII, followed by the related work in §VIII, and concluding remarks in §IX.

II. Dataset Overview

A. Datasets

Our dataset is provided by the monitoring unit of a DDoS mitigation company [35] utilizing both active and passive measurement techniques for monitoring attacks launched by certain malicious actors worldwide across America, Europe, Asia, Africa, and Australia, via the partnership with over 300 major ISPs globally. For active measurements and attribution, malware families (botnets) used in launching various attacks are reverse engineered, and labeled to a known malware family using best practices. Hosts participating in the given botnet, by either communicating with pieces of infrastructure infected by that malware family (e.g., the command and control) or launching the actual attack on the monitored targets are then enumerated and monitored over time, and their activities are logged and analyzed.

Our dataset is distinguished from datasets used in the prior work in one fundamental way: ground truth of $50,704$ DDoS attacks. In this paper, each DDoS attack examined in this paper is a ***verified attack*** by the target. Compared to the prior literature, this is a great improvement; e.g., Mao *et al.* [36] uses alarms and accept them as attacks for DDoS characterization (many of which could potentially be false alarms [37]).

As each attack was verified by the provider and customers, we are confident that those attacks reflect reality. The lack of data of such nature and size until now have delayed and undermined our understanding of DDoS patterns and attackers' behaviors. On the other hand, the dataset may not cover all corners of the globe, since the provider only works with major ISPs in the world.

B. Issues and Challenges

First, we note that DDoS attacks labeling is not in the scope of this work. However, we are provided with the attack information of both attackers and the victims from the dataset through a partnership with a DDoS mitigation operator using state-of-the-art practices for labeling and attribution of DDoS attacks. Nonetheless, the proposed model and related analyses are built upon this knowledge.

One might argue that such prerequisites are difficult to satisfy in the real-world context. However, the model's primary goal is to guide defense resources provisioning proactively instead of delivering forefront defense solutions reactively. With the help of the existing proposed attack identifications and filtering mechanisms such as [38], [39], [40], the model helps understand the attacking behaviors based on the historical data.

Though our model is centered around the data we use to capture DDoS behaviors, it flexibly exhibits features that capture different botnets. From previous explorations, we find that it is common for botnet families to have both geolocation and target preferences. Also, botnet families present periodic recruiting and dormancy patterns. All these suggest potential systematic and mathematical representation of the botnet-based attacking behaviors, which will be helpful to unveil the sophisticated strategies utilized by botmasters.

C. Collection Methodology

As each botnet evolves over time, new botnet generations are marked by their unique (MD5 and SHA-1) hashes. Traces of traffic associated with various botnets are collected at various sensors on the Internet, in cooperation with various Internet Service Providers (ISPs). Traffic logs are then analyzed to attribute and characterize attacks. The collection and analysis are guided by two general principles: 1) that the source of the traffic is an infected host participating in a botnet attack, and 2) the destination of the traffic is a targeted client, as concluded from eavesdropping on (or profiling) C&C of the campaign using analyzed malware samples.

By tracking temporal activities of 23 different known botnet families, the dataset captures a snapshot of each family every hour. In the rest of the paper we mainly focus on the 10 most active botnet families. Overall, the dataset covers about 7 months and spans the period from August, 2012 to March, 2013. There are 24 hourly reports per day for each botnet family. The set of bots or controllers listed in each report are cumulative over the past 24 hours. The 24-hour time span is measured using the timestamp of the last known bot activity and the time of the logged snapshot.

The analysis conducted over the data is high-level to cope with the volume of the traffic at peak attack times—on average there were 243 simultaneous verified DDoS attacks launched by the different botnets studied in this work. High level statistics associated with the various botnets and DDoS attacks are recorded every hour. In our data log, a DDoS attack is labeled with a unique DDoS identifier, corresponding to an attack by given DDoS malware family on a given target [21].

III. Feature Analysis and Extraction

A. Feature Space of DDoS attacks

In this section, we extract important attack features that will be used in the models. In the following, we pursue such an analysis focusing on the extraction of distinguishing features of DDoS attacks, including the magnitude of bots involved, durations of DDoS attacks, inter-launching time, activity level of bots and source distribution of bots.

1) Magnitude of bots: The number of bots associated with a DDoS attack is an essential feature, which we refer to as bots magnitude. In our dataset, each DDoS attack is uniquely identified by a DDoS ID. Each DDoS ID is associated with a timestamp, which represents the start time of the DDoS attacks. We first classify these DDoS attacks based on the botnet families used for launching them, since they are already labeled. In this way, multiple DDoS attack records belonging to the same family in our dataset could be ordered chronological; i.e., $\langle DDoS_1, DDoS_2, ..., DDoS_m\rangle$. Also, we extract all the IP addresses of bots involved in DDoS attacks from multiple records corresponding to the same DDoS attack. In this way, a DDoS attack could be represented by a series of bot IP addresses, i.e., $\langle ip_1, ip_2, \cdots, ip_n\rangle$. Since we assume that the magnitude of the attacking sources could be measured by the number of bots involved, which is reasonable and in line with the prior literature [36], each DDoS attack could be represented as a time series of numbers, which measure the attacking magnitudes at any recorded time. As a result, each DDoS attack itself is also a time series data since the magnitude of bots involved keeps changing over time.

2) DDoS attack turnaround time: Turnaround time by our definition is the total time taken between submission of an attack task for execution and the complete output to the attacker. The turnaround time of DDoS attacks is an important feature, and closely related to the strategies employed by the attackers. In attack scheduling, turnaround time includes two phases: waiting for execution and the execution time. We use duration of DDoS attacks to represent the execution time. Each DDoS attack in our dataset is associated with attribute $Duration$, which is the approximate length of time in seconds that the attack lasted. On the other hand, we use inter-launching time between consecutive DDoS attacks to simulate the waiting time. Once we arrange the relevant DDoS attacks in chronological order, the inter-launching time between DDoS attacks (inter-arrival time of attacks) is easily calculated. While multistage DDoS attacks are defined previously in the literature, for example in [22], we augment this definition to include attacks that happened consecutively within a timeframe of 30 seconds to 24 hours. This means that all DDoS attacks that happen within one day of each other and are targeted towards the same target are considered as multistage DDoS attacks, and as long as they were not launched at the same time. This range is obtained from analyzing the CDF of inter-launching time of any two consecutive DDoS attacks. This range covers most consecutive DDoS attacks without introducing much noise by chopping off the long head and tail of the distribution. This feature links multiple stages of DDoS attacks to one single tightly related attack.

3) Activity level of bots: We use the average number of DDoS attacks per day to capture this feature. Table I shows the average number of attacks per day, the total number of active days and coefficient of variation (CV) regarding to the daily number of attacks for each botnet family in our dataset. CV, also known as relative standard deviation (RSD), is a standardized measure of dispersion of a probability distribution. In our case, CV is used to measure the stability of bots activity levels, which is calculated as the ratio of the standard deviation to the mean (of the number of daily DDoS attacks). Lower CV values indicate higher stability of bots activity levels, and vice versa. Taking all three metrics in the table into consideration, we notice a spectrum of activity levels, where *DirtJumper* is most active while *AldiBot* is the least active family. Among all, *BlackEnergy*, *Pandora* and *DirtJumper* represent the most stably active families.

TABLE I
Activity level of bots.

Family	Avg. #/Day	# Active Days	CV
AldiBot	1.29	204	0.77
BlackEnergy	5.93	220	0.82
Colddeath	7.52	118	1.53
Darkshell	9.98	210	1.14
DDoSer	2.13	211	0.84
DirtJumper	144.30	220	0.77
Nitol	2.91	208	1.05
Optima	3.19	220	0.90
Pandora	40.08	165	1.27
YZF	6.28	72	1.41

4) Source distribution of bots: Beside the bots magnitude, the distribution of attack sources is an essential feature to characterize attacks. Since the IP information of bots in-

TABLE II
SUMMARY OF THE MAIN PARAMETERS.

Variable	Description
$A_{t_i}^f$	Botnet activities represented by attackers' botnet families at time t_i
$A_{t_i}^b$	Available bots (currently active) in the system at time t_i
$A_{t_i}^s$	Attacker source distributions according to the DDoS attack scheduling strategies of a botnet family at time t_i
T_l	Targets' geolocation information in the Internet represented by ASN
T_j^d	Valid duration time to take down the target from j-th DDoS attack
T_j^{ts}	Timestamps when j-th DDoS attack happen towards the target
$(D_{t_i}^b)_j$	Output of magnitude of bots at time t_i from j-th DDoS attack observed by the target
$(D_{t_i}^d)_j$	Output of remaining time left at time t_i of j-th DDoS attack observed by the target
D_{j+1}^{ts}	Timestamp when $j+1$-th DDoS attack happen at the target

volved in DDoS attacks and their dynamics are captured by our dataset [22], we directly utilize this information to evaluate the distributions. Since an IP-level low granularity characterization may not capture such a feature, we choose the AS-level characterization. For that, we map IP addresses to their corresponding ASNs (autonomous system numbers) using a commercial grade mapping dataset [41]. Accordingly, each DDoS attack represented by $\langle bot_1, bot_2, ..., bot_m \rangle$ is converted to $\langle AS_1, AS_2, ..., AS_m \rangle$. Then, we further line up all the DDoS attacks launched in chronological order to obtain a time series of AS distribution data.

B. Modeling Features

In order to pursue this modeling effort, we define a few variables to capture these features. Table II outlines the important variables, which we define in the following. In modeling DDoS attacks, we use three groups of parameters. The first group describes the botnet states at the attackers' side, including the activity levels of bots and their dynamic distributions based on their locations. The second group illustrates target affinity to attacks based on their static attributes, e.g., locations. Finally, the third group contains the results generated by our model, which are used as feedback to correct our modeling results. In Table II, attributes used to model attackers, namely $A_{t_i}^f$, $A_{t_i}^b$ and $A_{t_i}^s$, have a time aspect to them, and are sorted chronologically. On the other hand, all attributes used to characterize targets, namely T_l, T_j^d and T_j^{ts} are independent of time. Thus, all output variables of our model contain time information as well, which defines the dynamics in DDoS attacks. Next we describe the three groups of variables used for modeling DDoS attacks.

1) Botnet state: We first define parameters to describe the activity levels of botnet families. First of all, the activity level could be captured by the number of DDoS attacks launched per day by the given family by far. Thus we define $A_{t_i}^f$ to describe the attack activities and it could be calculated by a linear combination of its history observations. The weights are assigned dynamically using the training process of the model. On the other hand, $A_{t_i}^b$ representing the total number of currently active bots in this botnet family, also describes activity level of a botnet family. This variable could be represented by percents of active bots in all historic observations. Variable $A_{t_i}^s$ describes the linearity of relationship between attacking bots and the previous observations. Since the bots involved in an attack may rotate or shift, the source distribution is also calculated based on the previous observations, whose weights are obtained from learning.

Finally, we note that these three variables are not completely independent on each other: the number of current active bots depends on its own botnet family F_b, the activity level and the distributions of these bots.

2) Target affinity: This group of variables describes the information related to targets. In particular, it captures the affinity of attackers to certain targets. T_l represents the ASN of a target. The value of this variable is determined once a DDoS attack is detected, since it is tied to the IP information of a target. The affinity of the targets may also influence the duration of a DDoS attack. Because the duration T_j^d of a DDoS attack towards a target depends on not only the historical data of the durations of previous attacks on that target (or targets defined within the same network; AS-level), but also the number of active bots A_t^b when the j-th DDoS attack happens. Notice that the durations of these attacks do not necessarily have linear relationships with each other.

In this work, we decompose and represent the timestamp of DDoS attacks into two parts: day and hour $T_j^{ts} = (T_j^{day}, T_j^{hour})$. Such a choice of decomposition is motivated by various reasons, which are outlined in the following.

First, the time when DDoS attacks were launched is usually determined by botmasters. They usually choose time based on their bot activity patterns to perhaps minimize detection and blocking. Second, by confining the variable into a closed interval range, e.g $[0, 24)$ or $[1, 31]$, it may reveal some patterns of DDoS attacks for predictors that makes the learning of the attack perhaps possible (equivalent to aggregating the attack on daily and hourly basis). Third, from the defense perspective, it makes sense to accommodate defense deployments with the modeling and prediction results dynamically during the course of a day or a month.

3) Modeling output: The ultimate goal of our modeling is to make predictions on the behavior of DDoS attackers, and would result in predicting DDoS attacks and their features. Such predictions would ultimately help refine the defense posture, by adjusting resources to minimize the damage of an adversary. Thus, this group of variables serves as both output results and feedback to our model.

The magnitude of bots involved in DDoS attacks, $(D_{t_i}^b)_j$ is an important factor in DDoS attacks since it indicates the magnitude of damages. The magnitude of bots involved in a DDoS attack launched at time t_i is dependent on the location of targets T_l, the number of active bots observed by time t_i, $T_{t_i}^b$ and the linear combination of all previous observations before attack D_j. Another feedback variable in our model is the timestamps (represented by hour and day) of the DDoS attacks. In our model, they are determined by the target location, the current number of active bots in a family, and result on timestamps of DDoS attacks that happened before.

Finally, the duration of a DDoS attack is constrained by several factors. The attack duration depends on the remaining time of an DDoS attack and it is labeled as time series data since it may change during the course of DDoS attacks due to a change in the underlying features of the attack. For example, if bots involved in an attack were taken down, the attack cannot be carried on. However, for simplification, we convert this variable to a scalar value by modeling the total duration of DDoS attacks annotated by D^d_j. This value takes into consideration some potential changes during the DDoS attacks. This is achieved by correlating the duration with the linear results of previous DDoS attacks observed and the average magnitude of involved bots in previous DDoS attacks. In addition to these two principal variables, the location of the target and the timestamp of when DDoS attack D_{j-1} happened also contribute to the modeling of the attack duration.

C. Temporal, Spatial and Spatiotemporal

Based on the above discussions, some variables are time-related while others are not. Any meaningful model to characterize DDoS attacks by modeling should be comprehensive to capture all of those aspects. Since we have three types of variables, our model is composed of three components. The first component is used to capture the dynamic time-series relationship between multiple variables, the second component is concerned with the spatial relations represented by the targets, and the last component is to combine the two previous components to represent the outcomes of our comprehensive model. We present those three components in the three subsequent sections.

Models can be validated in two ways: *goodness of fit* of the model and *quality of prediction*. The goodness of fit of a statistical model describes how well it fits a set of observations, whereas prediction leverages statistics in the model to predict outcomes concerning observations. In evaluating the models proposed in this paper, we focus particularly on the latter method. In particular, we measure the power of models in predicting certain features of the DDoS attacks. To verify the accuracy of our models using their predictions, we split the data into two parts: $40,563$ attacks for training and the rest $10,141$ attacks for testing.

There are two competing concerns with the dataset split: the parameter estimates would have greater variances with less training data. On the other hand, with less testing data, the performance statistic might have greater variances. We choose 80% of our data to train the model while minimizing the possibility of overfitting given the scale of our dataset. The data in the testing set has no effect on training. Thus, it provides an independent measure of performance during and after training.

IV. Temporal Modeling

A. Model Construction

The temporal model is utilized for capturing the time series relationships between variables of a DDoS attack and to predict the DDoS's happening in the future. There are three variables related to this model: $A^f_{t_i}$, $A^b_{t_i}$ and $A^s_{t_i}$. These variables provide the initial time information to the entire model. We describe them below.

1) Activity level: This variable describes the average number of DDoS attacks per day by a specific botnet family. A feature based on this variable is defined as:

$$A^f_{t_i} = \frac{\sum_{t_i \in T} N_{DDoS}}{\sum t_i \in T}, \tag{1}$$

where N_{DDoS} is the total number of DDoS attacks that occurred thus far for botnet family f at time t_i.

2) Attack magnitude: The total number of active bots at time t_i is defined as $A^b_{t_i} = N^{active\_bots}_{t_i}$. Among all, $N^{active\_bots}_{t_i}$ represents the number of unique IP addresses involved in DDoS attacks. In our model, we assume that IP addresses map to bots in one-to-one fashion, which is accepted in the prior work (c.f. Mao *et al.* [36]). For different botnet families, the scale of their harms varies. As a result, the absolute value (magnitude) of active bots may bias our modeling results. To minimize such bias, we further calculate the number of bots as:

$$A^b_{t_i} = \frac{N^{active\_bots}_{t_i}}{\sum_{t=1}^{t_i} N^b_t}, \tag{2}$$

where N^b_t is the total number of bots by a botnet family by time t_i.

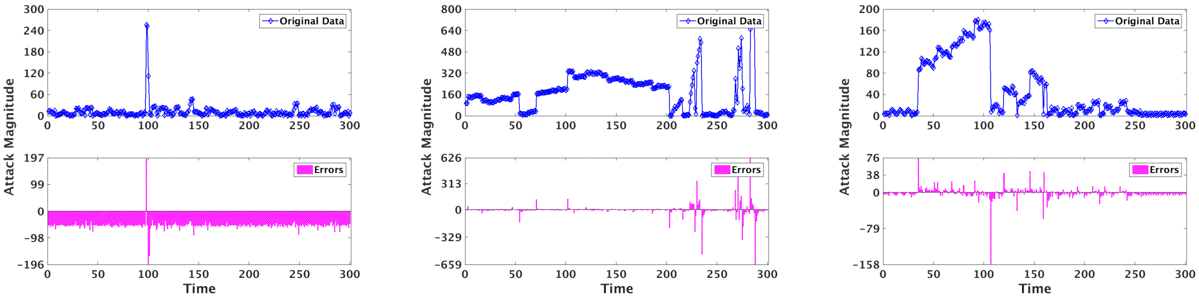
3) Source distribution: To quantify this variable in our model, we need to first scale it based on actual measurements. Inspired by the *Silouette coefficient* [42] in validating consistency within clusters of data, we calculate the source distribution as:

$$A^s_{t_i} = \frac{\sum_{j=1}^{n} I^{AS_j}_{t_i}}{DT_{t_i}}. \tag{3}$$

Eq. (3) is two parts: *intra-AS* distributions and *inter-AS* distributions in the numerator and denominators, respectively. $I^{AS_j}_{t_i}$ represents the *intra-AS* distribution for AS_j, which is calculated as:

$$\begin{aligned} I^{AS_j}_{t_i} &= \frac{N^{AS_j}_{t_i}}{N_{AS_j}}, \\ DT_{t_i} &= \frac{2 \times \sum_{k=j}^{n} \sum_{j=1}^{n} (DT^{AS_k}_{t_i} - DT^{AS_j}_{t_i})}{n \times n - 1}. \end{aligned} \tag{4}$$

The first part in Eq. (4) captures the evaluation of *intra-AS* distribution. $N^{AS_j}_{t_i}$ represents the number of bots located in AS_j, while N_{AS_j} represents the total number of available IP addresses in AS_j. On the other hand, the second part in Eq. (4) measures the *inter-AS* distribution, which captures the average distance between AS_k and AS_j at time t_i. For the inter-AS distance $(DT^{AS_k}_{t_i} - DT^{AS_j}_{t_i})$, we develop a tool to



(a) *BlackEnergy*

(b) *DirtJumper*

(c) *Pandora*

Fig. 1. Prediction of attacking magnitudes.

infer AS relationship from one or more routing tables provided by Route Views [43].

The underlying algorithm utilized in our tool is based on the work of Gao *et al.* [44]. Using the relationships between ASes, we could further infer the path from one AS to another. Furthermore, we can calculate the distance between them (in hops). As a result, the inter-AS distribution is measured by the average hop distances between ASes involved in DDoS attacks. In this way, the more bots are located in fewer ASes, the larger $I_{t_i}^{AS_j}$ and the smaller DT_{t_i}, thus resulting in larger $A_{t_i}^{s}$ value, and vice versa.

4) Model composition: With these three features, we further build models on top of them. All three variables $A_{t_i}^{f}$, $A_{t_i}^{b}$ and $A_{t_i}^{s}$ are represented as time series data: $\{A_t^f\} = (A_{t_1}^f, A_{t_2}^f, A_{t_3}^f, ..., A_{t_n}^f)$, where time series analysis of each of those variables involves modeling the series as a function of its past observations and errors. Specifically, the temporal model is based on a linear regression (LR), which models the local DDoS attack regression process, since local DDoS attacks are arranged in chronological order.

To this end, we choose autoregressive integrated moving average (ARIMA) model, which is the most general class of models for time series data [45], [46]. It can capture linear time series correlations. In ARIMA, there are two fundamental building blocks: the autoregressive (AR) model and the moving average (MA) model. The forecast in AR is a function of its past observations, while in MA, it models the function of past errors to make corrections. So with ARIMA, the model of $A_{t_i}^{f}$ would be presented as:

$$A_{t_i}^{f} = \sum_{j=1}^{p} \phi_j \times A_{t-j}^{f} + \sum_{j=0}^{q} \theta_j \times e_{t-j}^{f}. \tag{5}$$

B. Prediction Results

The performance of temporal model is validated by testing its power in predicting variables associated with DDoS attacks. Using the temporal model we predict DDoS attacks. The predicted results are shown in Figure 1 for the 3 most active families (*BlackEnergy*, *DirtJumper* and *Pandora* from Table I). In these figures, the x-axis represents the time sequence; the y-axis represents the magnitude of the attacking sources. There are two subfigures in each figure, where the top one shows the ground truth data while the bottom displays the errors of the prediction results. Longer bar means larger error and vice versa.

One observation from these figures is that for *Pandora* and *DirtJumper*, the predicted results are almost identical to the ground truth. For *BlackEnergy*, despite the observed difference between the ground truth and the prediction groups, the trends are identical for them since the errors present in both are a constant. However, constant errors like this do not affect the prediction results on the scale of DDoS attacks. To avoid over-provisions of the defense resources, the accuracy of the modeling needs to be improved, which will be addressed later by the spatiotemporal model.

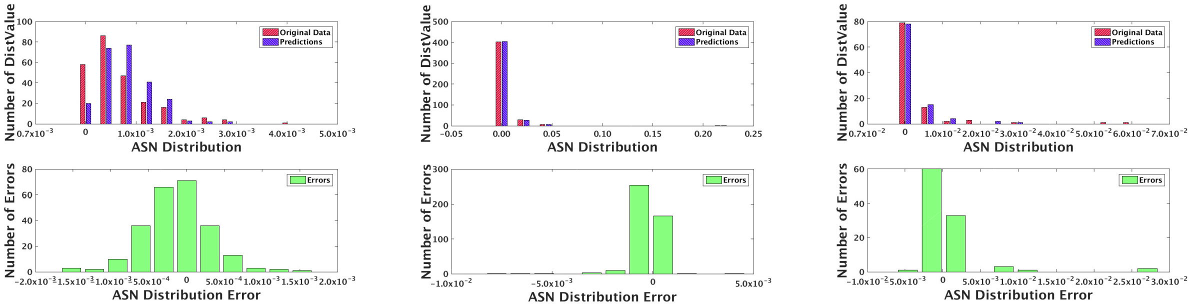
V. Spatial Modeling

All target-related variables characterize DDoS attacks in the same network region (AS-level). Thus, we do not treat these DDoS attacks as time series data though they naturally could be sorted in chronological order based on their timestamps. Consider each participating bot as a neuron in a botnet, and bots collaborate with each other to complete the task and there are nonlinear relationship between them, we choose neural network (NN) to model its spatial behavior features. Since different DDoS attacks may be launched by different botnet families, they may have weak or nonlinear relationships with each other. Thus, if we replace $\sum$ in (5) with a nonlinear activation function, a nonlinear auto-regressive (NAR) model is created.

A. Model Construction

The features we use as inputs to the spatial model are T_l, T_j^d and T_j^{ts}. For T_l, to obtain this variable we build a map to convert IP addresses of bots to ASNs, based on the AS information obtained from the `whois` database [41]. On the other hand, T_j^d is calculated as T_j^{end} - T_j^{start} to represent the valid durations of the DDoS attack. Finally, the feature T_j^{ts} is decomposed into $\langle T_j^{day}, T_j^{hour} \rangle$ as discrete values.

Besides those variables, the time between attacks is also needed for presenting the frequency of DDoS attacks that happen in a given network to help modeling T_j^{ts}. This time is calculated as $T_t^i = T_{j+1}^{ts} - T_j^{ts}$. With these features, we have two options for the spatial model: either using time-series or non-time-series models. We use the time-series model.



(a) *BlackEnergy*

(b) *DirtJumper*

(c) *Pandora*

Fig. 2. Prediction of attacking source distributions.

The spatial model consists of three layers: input, hidden and an output. These layers are connected to each other with different connection weights.

In our model, we use only one hidden layer to construct the spatial model in order to simplify the performance optimization, with the output evaluated as:

$$T^d_{j+1} = f(T^d_j, T^d_{j-1}, T^d_{j-2}, \dots, T^d_{j-q}) + \epsilon_j, \tag{6}$$

$$\epsilon_j \sim N(0, \sigma^2). \tag{7}$$

In (6), q represents the number of delays, which means that T^d_{j+1} is modeled as a nonlinear autoregression function of the past q values plus a normal error term. Three transfer functions are most commonly used for multilayer networks, including Log-Sigmoid Transfer Function, Tan-Sigmoid Transfer Function and Linear Transfer Function. For the hidden layer, the transfer function has to be nonlinear functions to avoid linear only separable solutions. As a result, we choose the default Tan-Sigmoid Transfer Function [47] to be the activation function.

Except with the time series model, if we consider both T^d_j and T^{ts}_j as non-time-series features, we have to introduce other attack-related features, namely $A^b_{t_i}$ and $A^s_{t_i}$. Otherwise, we need to use all T^d_t, $t \leq t_i$ as different features for training. However, the latter model is not particularly useful given the nature of these variables.

An important parameter for the NAR model is the number of hidden nodes, which is used to transform the inputs into data that the output layer can use. For each dataset by any botnet family, we need to find the optimal parameters for the number of delays as well as the number of hidden nodes. A grid search technique [48] was utilized to accomplish this.

B. Prediction Results

For the attackers' source distribution, we first split the DDoS attacks based on the targets' ASN and then model the chronologically ordered DDoS attacks within each network (AS-level). Finally, the modeling results of *BlackEnergy*, *DirtJumper* and *Pandora* family are shown in Figure 2. In each of these figures, there are two subfigures. The top one compares the attacker ASN distribution calculated using the ground truth data and predictions; and the bottom one displays the error distributions. Clearly, the distributions of predicted results for *DirtJumper* and *Pandora* are almost 100% accurate. For *BlackEnergy*, though the distribution looks slightly different, most prediction results are still accurate by observing the error distribution. Such results suggest that the AS-level distribution of attack sources could be accurately predicted as well. Such capability could further facilitate effective defense mechanisms via early DDoS attack detections, which could be achieved by evaluating the entropy of AS distributions over all concurrent connections.

VI. Spatiotemporal Modeling

With temporal and spatial models, we can obtain their output that holds over the entire (temporal or spatial) feature-space. In practice, temporal and spatial features of attacks also interact with each other, and it will be much more difficult to build a global model to capture the interactions. As such, an alternative approach to nonlinear regression is to partition the data space into smaller regions recursively, where the interactions are more manageable.

A. Model Construction

The spatiotemporal models are built on top of the smaller regions using simpler learning models, like the linear regression. To combine these two parts together, we use Regression Tree (RT) [49] (as a combination of the partitioning and linear regression).

In RT, each terminal node (leaf) represents a cell of the partition. To explain our model, let us first consider the case of a region R of a feature space described by five variables $A^f_{t_i}$, $A^b_{t_i}$, $A^s_{t_i}$, T^b_j and T^d_j. We want to model T^d_j based on R. R can be partitioned into three partitions R_1, R_2 and R_3:

$$R_1 : T^d_j = \beta A^b_{t_i} + \mu T^b_j, \tag{8}$$

$$R_2 : T^d_j = \beta' A^b_{t_i} + \mu' T^b_j, \tag{9}$$

$$R_3 : T^d_j = \delta A^s_{t_i} + \mu'' T^b_j, \tag{10}$$

such that $A^f_{t_i} \leq \alpha$ and $A^b_{t_i} \geq \lambda$ fall into category R_2. Each leaf node is attached to a simple model, in this case a multivariate linear model (MLR). In this model, the presence of T^b_j in both R_1 and R_3 indicates that this variable is relevant both when $A^s_{t_i} < \beta$ and $A^f_{t_i} \geq \alpha$ although its influence on the dependent variable T_j^d could be very different for the two regions. Accordingly, the effect of variable T_j^b is local since it can be properly modeled by considering $A_{t_i}^f < \alpha$. In the construction of a model tree, the main problem to solve is to choose the best partition of a region in the feature space. In our model, we use Classification And Regression Tree (CART) [49].

Once we built the tree structure, for each leaf node we can use simpler models to describe the correlations between variables (i.e., outputs of temporal and spatial models) as indicated in Eq. (8). In our model, we use the multivariate linear regression (MLR) model to connect independent and dependent variables. Thus, in the spatiotemporal model, the relations between multiple relevant variables could be captured in a comprehensive way.

B. Prediction Results

The previous prediction results are shown at the level of a botnet family. However, the ultimate goal of our models is to make predictions on specific DDoS attacks. In most operational scenarios, however, the difficulty of predicting a specific target stems from the lack of historical data to make the predictions accurately. As a result, the spatiotemporal model might help address this issue. From the target's standpoint, the most important and relevant features include magnitude of bots involved during the DDoS attacks, the time when the DDoS attack happen and how long it lasts.

To train the spatiotemporal model, we use two sets of DDoS attacks for each target. We assume that the target has the accessibility to the observations of 1) part of DDoS attacks happened within the same AS area and 2) part of DDoS attacks happened anywhere recently. This assumption is reasonable, especially when cloud-based DDoS mitigations are being widely deployed and utilized. The security service providers could share such information with customers or generate the predictions themselves and deliver the results back in response to DDoS attacks [50], [51]. To simulate such a scenario, we use 10 historical attacks for each group for each target, extract the relevant features and then feed these features into the spatial and the temporal models, respectively. After that, we use the results generated by these two models to train a Regression Tree on the feature and validate the predicted results with the next DDoS attack observed by this target.

An important feature we examine is the time when a DDoS attack happens. As discussed, instead of predicting the intervals in seconds, we use both the hour and day parts of the timestamp of the attack. In this case, we need to train a Regression Tree to combine the temporal and the spatial prediction results. In the constructed tree, we have node N_{tmp} representing the hourly prediction results generated by the temporal model, node N_{spa} representing the hourly prediction results generated by the spatial model, and node N_{int} representing the intervals in seconds for the prediction results generated by the temporal model. The hour when DDoS attacks happen is strongly correlated with the above features. To avoid overfitting, we prune the tree to keep only 88% of the original standard deviations. In the unpruned tree, the time is determined by the average magnitude of bots as well. The final results are shown in Figure 3.

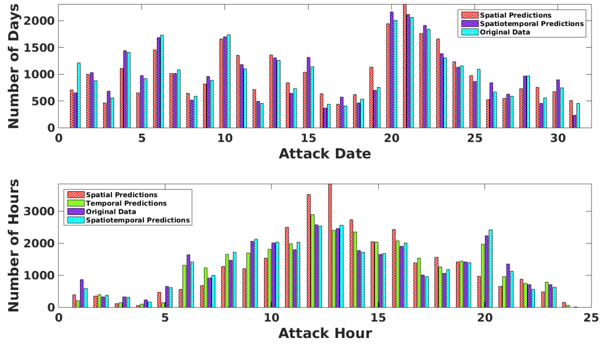


Fig. 3. Spatiotemporal predictions for DDoS attack timestamps.

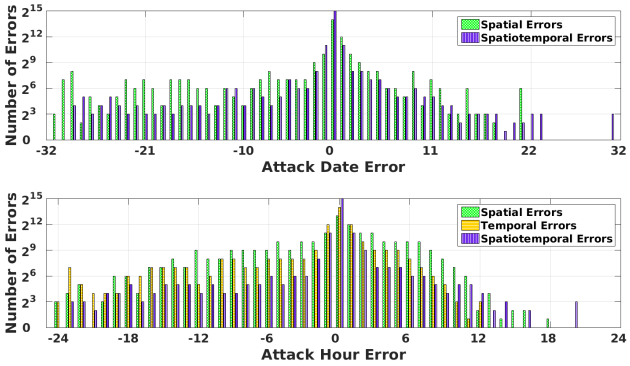


Fig. 4. Spatiotemporal prediction error distributions.

Two groups of data are presented in this figure: distributions of attack date for all models except for the temporal model shown on top, and the distributions of attack hour for all models down (different colors of bars represent different models). We exclude the temporal model in this case because it does not help with the prediction of a specific target. As a result, only results generated by the spatial and the spatiotemporal models are shown in this figure. From the figure, we have several observations. First, we clearly observe that the spatiotemporal model generates much better results than the other two models for both the date and hour, since its output is closer to the ground truth data. Second, the temporal model preforms slightly better than the spatial model in predicting attacking hours. This is further visually verified in Figure 4. The two subfigures in this figure show the error distributions comparison of all available models. Notice that all the values on the y-axis are in log scale.

Clearly, the spatiotemporal model outperforms other models in both cases. We calculate the RMSE for all three predictions to compare their performance. In predicting the hour variable, for the spatial model the RMSE is 5.0 hours. On the other hand, for the temporal model, the RMSE is 3.82 hours, while it is 1.85 hours for the spatiotemporal model. While in date predictions, RMSE is 5.17 days for the spatial model and 2.72 days for the spatiotemporal model. To summarize, the spatiotemporal model greatly improves the accuracy of timestamp predictions. Combining these two predictions will lead to an accurate prediction on when the next DDoS attack will happen.

In summary, DDoS attacks are accurately modeled from three different perspectives: attack time, attack duration, and magnitude of bots. For a specific target, this information is critical to help deploy defenses accordingly. Further, with the help of the spatiotemporal model, the accuracy of modeling could be greatly improved.

VII. Discussion

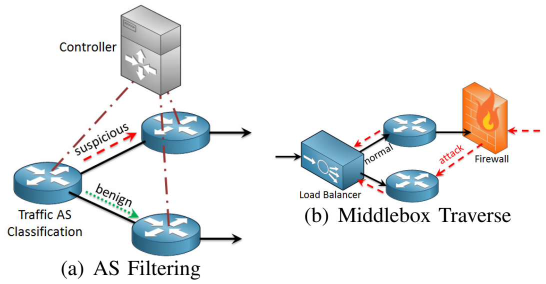
A. Comparison

We have shown that DDoS attacks could be accurately characterized using our models. One may argue that simple models could also work well. For example, one may advocate a simpler approach in which prediction outcomes are the same as (or the mean of) previous observations. To test such an argument, we compare our models, both temporal and spatial, with these two simple predictions, which are referred as *Temporal/Spatial, Always Same* and *Always Mean*, respectively.

Three features are used for comparisons: the magnitude of bots, the durations of the DDoS attacks and the ASN distribution of bots. We compare the RMSE of different modeling outputs on the five most active botnet families. We observe that the *Temporal/Spatial* model always generates better prediction results for all three features. In some cases, however, the *Always Same* and *Always Mean* models generate biased results that are almost useless. The failure of these two simple models highlights the sophisticated and dynamic strategies utilized by the attackers. Thus, without an in-depth understanding of the attackers, it is difficult, if not impossible, to predict how DDoS attacks evolve utilizing intuitive and simple observations. The comparisons also demonstrates that our models manage to strike a balance between the performance and accuracy.

B. Use Cases

One can utilize the predictive power of a comprehensive model to guide the deployment of defense mechanisms. More specifically, with the knowledge of the time and the scale of the next DDoS attack, it is possible to proactively deploy defense resources that would effectively thwart the attacks. Such proactive defenses guided by our predictive models are indirectly more cost effective, since they provide a better utilization of limited defense resources. Furthermore, ASN distributions provide information of where the attacking traffic is most likely to come from. ASN distributions also indicate the possible malware utilized by botnets due to the location affinity property of botnet families [21]. As a result, using predicted attack source distribution, filtering resources could be mobilized on the fly closer by the adversary, adversaries could be attributed to certain malware families that could be contained by rapidly updating antivirus signature and ISPs filtering middleboxes, among other potential techniques.



(a) AS Filtering

(b) Middlebox Traverse

Fig. 5. Use cases of different attack scenarios

With the advancement of networking technologies, novel techniques such as Software-Defined Networking (SDN) enables more flexible and dynamic deployment of security mechanisms. We will discuss three use cases with our model applied in the following to show how security mechanisms could be improved.

1) AS-based filtering: In traditional networking, it is difficult, if not impossible, to include detailed metadata for packet classifications in routers. To accommodate future communication protocols and facilitate network virtualization functions, recent designs of SDN compatible devices have expanded their capabilities in packet classifications. As a result, it is effortless to include classification based on AS-related information, especially with the SDN-supported switches. As such, our model could run in the control plane to help differentiate attack flows based on their AS distributions as shown in Figure 5(a). All the traffic belonging to the AS that falls into the attacking source ASes will be forwarded along different route path for further examinations. Such classification could be achieved at any ingress SDN-compatible routing devices.

2) Middlebox traverse: Middleboxes are prevalent in networking configurations to provide auxiliary security services. Policies for different components are composed for security enforcement. The order of processing of different middleboxes plays an important role in this composition. An illustration is provided in Figure 5(b). In normal cases, where networks are not under attack, the traffic traverses the load balancer before the firewall for better throughput so that only suspicious traffic get scrutinized. While under DDoS attacks, the traffic will reverse its path to get processed by the firewall before the load balancer to make sure that the packets will not be modified to evade detection. To employ such mechanisms dynamically and "gracefully", predictions of the time when DDoS attacks are going to happen is necessary to minimize service interruptions, which is provided by our model.

In summary, networking behaviors should change adaptively as the traffic varies as suggested in [52] to enable more flexible security mechanisms. With the advancement of network technologies, it is becoming easier to manage network resources dynamically based on given policies. Our proposed model, on the other hand, provides guidance and references for such management to achieve more favorable security solutions. In this way, the defense could utilize traffic engineering techniques to make better use of the existing modules to address security challenges based on the actual, modeled, and predicted behaviors of the adversary.

VIII. Related Work

As botnet-based DDoS attacks prevail and gradually dominate the underground market, they evolve in both complexity and organization capacity. Motivated by that, a large array of research work focus on DDoS detection and mitigation techniques, which have been summarized in [53]. Yet understanding the strategies and behaviors of the attackers is the key to defending against these attacks.

The prior work on DDoS behavior modeling for defenses includes [54], [55], [15], and falls into two classes: malicious activity detection and mitigations and dynamic predictions. Our work falls in the second class. A lot of research has tackled the challenges in the first phase, from attack detection and mitigation to attack source identification and taking down. Fedynyshyn *et al.* [54] enumerated independent communication features for building botnet detection models. Bilge *et al.* [55] utilized supervised machine learning techniques on several groups of features to identify command and control channels. Rossow *et al.* [15] proposed a formal graph model to evaluate the intrinsic vulnerabilities of P2P botnets. Other work on botnet C&C modeling and detections also includes [56], [57], [58], [59], [60]. We also conducted a measurement study on some of the most active botnets on the Internet to examine and compare the attacking capabilities of different families [61].

Qin *et al.* [32] used statistical tools to analyze the security alerts, where they mainly focused on attack scenarios by correlating the alerts and finding their causality relations. Wang *et al.* [62] also used attack graphs for analyzing the alerts and making predictions. Their work differs from previous work in using a queue graph instead of the timing windows to analyze recent alerts, requiring less memory and speeding up the attack correlation tasks. One of the limitations of earlier work is that static analysis of detection alerts was used to make predictions.

IX. Conclusion

Despite tremendous defense efforts, DDoS attacks are still prevalent. In this paper, we made an exploratory attempt to understand the botnet-based DDoS attacks via a modeling approach. Our goal is to accurately predict their occurrence and associated features of botnet based DDoS attacks based on historical information. For this purpose, we have constructed models to capture the temporal and spatial features of DDoS attacks and integrate them into a spatiotemporal model. Based on more than 50,000 confirmed DDoS attack workload during a seven-month period that was collected from services on the Internet globally, we have validated our model and demonstrated that our model can predict the DDoS attacks with high accuracy in terms of the magnitude, duration, inter-launching time, and location.

Acknowledgement

We thank the anonymous reviewers for their comments. This work was supported in part by the Global Research Lab. (GRL) Program of the National Research Foundation (NRF) funded by Ministry of Science, ICT (Information and Communication Technologies) and Future Planning (NRF-2016K1A1A2912757). This work was also supported by ARO grant W911NF-15-1-0262, NIST grant 70NANB16H166 and NSF grant CNS-1524462.

References

[1] R. Rasti, M. Murthy, N. Weaver, and V. Paxson, "Temporal lensing and its application in pulsing denial-of-service attacks," in *Proc. of IEEE S&P*, 2015.

[2] S. K. Fayaz, Y. Tobioka, V. Sekar, and M. Bailey, "Bohatei: Flexible and elastic ddos defense," in *Proc. of USENIX Security*, 2015.

[3] T. Vissers, T. van Goethem, W. Joosen, and N. Nikiforakis, "Maneuvering around clouds: Bypassing cloud-based security providers," in *Proc. of ACM CCS*, 2015.

[4] C. Rossow, "Amplification hell: Revisiting network protocols for ddos abuse," in *Proc. of NDSS*, 2014.

[5] B. Agarwal, A. Akella, A. Anand, A. Balachandran, P. Chitnis, C. Muthukrishnan, R. Ramjee, and G. Varghese, "Endre: An end-system redundancy elimination service for enterprises." in *Proc. of NSDI*, 2010.

[6] A. Welzel, C. Rossow, and H. Bos, "On measuring the impact of ddos botnets," in *Proc. of EuroSec*, 2014.

[7] J. Clark and P. C. van Oorschot, "Sok: SSL and HTTPS: revisiting past challenges and evaluating certificate trust model enhancements," in *Proc. of IEEE S&P*, 2013.

[8] J. Ma, W. Yang, M. Luo, and N. Li, "A study of probabilistic password models," in *Proc. of IEEE S&P*, 2014.

[9] U. Rührmair and M. van Dijk, "Pufs in security protocols: Attack models and security evaluations," in *Proc. of IEEE S&P*, 2013.

[10] M. C. Tschantz, A. Datta, and J. M. Wing, "Formalizing and enforcing purpose restrictions in privacy policies," in *Proc. of IEEE S&P*, 2012.

[11] M. Polychronakis, "Ghost turns zombie: Exploring the life cycle of web-based malware." in *Proc. of USENIX LEET*, 2008.

[12] S. Jana and V. Shmatikov, "Abusing file processing in malware detectors for fun and profit," in *Proc. of IEEE S&P*, 2012.

[13] G. Maier, A. Feldmann, V. Paxson, R. Sommer, and M. Vallentin, "An assessment of overt malicious activity manifest in residential networks," in *Proc. of DIMVA*, 2011.

[14] B. B. Kang, E. Chan-Tin, C. P. Lee, J. Tyra, H. J. Kang, C. Nunnery, Z. Wadler, G. Sinclair, N. Hopper, D. Dagon *et al.*, "Towards complete node enumeration in a peer-to-peer botnet," in *Proc. of ACM ASIACCS*, 2009.

[15] C. Rossow, D. Andriesse, T. Werner, B. Stone-Gross, D. Plohmann, C. J. Dietrich, and H. Bos, "Sok: P2pwned-modeling and evaluating the resilience of peer-to-peer botnets," in *Proc. of IEEE S&P*, 2013.

[16] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu, "Design and analysis of a social botnet," *Computer Networks*, vol. 57, no. 2, pp. 556–578, 2013.

[17] X. Wang and M. K. Reiter, "A multi-layer framework for puzzle-based denial-of-service defense," *IJIS*, vol. 7, no. 4, pp. 243–263, 2008.

[18] M. Casado, P. Cao, A. Akella, and N. Provos, "Flow-cookies: Using bandwidth amplification to defend against ddos flooding attacks," in *Proc. of IWQoS*, 2006.

[19] J. Czyz, M. Kallitsis, M. Gharaibeh, C. Papadopoulos, M. Bailey, and M. Karir, "Taming the 800 pound gorilla: The rise and decline of ntp ddos attacks," in *Proc. of IEEE IMC*, 2014.

[20] D. Andriesse, C. Rossow, and H. Bos, "Reliable recon in adversarial peer-to-peer botnets," in *Proc. of IMC*, 2015.

[21] A. Wang, A. Mohaisen, W. Chang, and S. Chen, "Delving into internet ddos attacks by botnets: characterization and analysis," in *Proc. of IEEE DSN*, 2015.

[22] A. Wang, A. Mohaisen, W. Chang, and S. Chen, "Capturing ddos attack dynamics behind the scenes," in *Proc. of DIMVA*, 2015.

[23] A. Wang, A. Mohaisen, W. Chang, and S. Chen, "Measuring and analyzing trends in recent distributed denial of service attacks," in *Proc. of WISA*, 2016.

[24] E. Kline, M. Beaumont-Gay, J. Mirkovic, and P. Reiher, "Rad: Reflector attack defense using message authentication codes," in *Proc. of ACSAC*, 2009.

[25] M. Natu and J. Mirkovic, "Fine-grained capabilities for flooding ddos defense using client reputations," in *Proc. of ACM LSAD*, 2007.

[26] M. Xie and H. Wang, "A collaboration-based autonomous reputation system for email services," in *Proc. of IEEE INFOCOM*, 2010.

[27] M. Antonakakis, R. Perdisci, D. Dagon, W. Lee, and N. Feamster, "Building a dynamic reputation system for dns." in *Proc. of USENIX Security*, 2010.

[28] Arbor Networks, "Worldwide Infrastructure Security Report," http://bit.ly/1Tiqacw, 2015.

[29] Verisign Inc, "Annual Report 2014," http://bit.ly/1qm3eyA, 2014.

[30] Neustar, "Annual Report 2014," 2014.

[31] H. Du and S. J. Yang, "Probabilistic inference for obfuscated network attack sequences," in *Proc. of IEEE DSN*, 2014.

[32] X. Qin and W. Lee, "Attack plan recognition and prediction using causal networks," in *Proc. of ACSAC*, 2004.

[33] W. Zang, P. Liu, and M. Yu, "How resilient is the internet against ddos attacks?–a game theoretic analysis of signature-based rate limiting," *Int'l J. of Intel. Control and Sys*, vol. 12, no. 4, pp. 307–316, 2007.

[34] G. Yan, R. Lee, A. Kent, and D. Wolpert, "Towards a bayesian network game framework for evaluating ddos attacks and defense," in *Proc. of ACM CCS*, 2012.

[35] Team Cymru, "http://www.team-cymru.org/."

[36] Z. M. Mao, V. Sekar, O. Spatscheck, J. van der Merwe, and R. Vasudevan, "Analyzing Large DDoS Attacks using Multiple Data Sources," in *Proc. of ACM LSAD*, 2006.

[37] Y. Chen, K. Hwang, and W.-S. Ku, "Collaborative detection of ddos attacks over multiple network domains," *IEEE TPDS*, vol. 18, no. 12, pp. 1649–1662, 2007.

[38] R. Sahay, G. Blanc, Z. Zhang, and H. Debar, "Towards autonomic ddos mitigation using software defined networking," in *Proc. of NDSS workshop on SENT*, 2015.

[39] A. Alwabel, M. Yu, Y. Zhang, and J. Mirkovic, "Senss: observe and control your own traffic in the internet," *ACM SIGCOMM CCR*, 2015.

[40] J. Li, S. Berg, M. Zhang, P. Reiher, and T. Wei, "Drawbridge: software-defined ddos-resistant traffic engineering," in *ACM SIGCOMM CCR*, 2014.

[41] Whois, "Whois.net," https://www.whois.net/.

[42] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[43] Route Views, "Route views project," http://www.routeviews.org/.

[44] L. Gao and F. Wang, "The extent of as path inflation by routing policies," in *Proc. of IEEE GLOBECOM*, 2002.

[45] J. D. Hamilton, *Time Series Analysis*. Princeton University Press, 1994, vol. 2.

[46] A. Mohaisen, M. Bhuiyan, and Y. Labrou, "Name server switching: Anomaly signatures, usage, clustering, and prediction," in *Proc. of WISA*, 2014.

[47] D. L. Elliott, "A better activation function for artificial neural networks," University of Maryland, Institute for Systems Research Technical Reports TR-93-8, 1993.

[48] H. Larochelle, D. Erhan, A. Courville, J. Bergstra, and Y. Bengio, "An empirical evaluation of deep architectures on problems with many factors of variation," in *Proc. of ICML*, 2007.

[49] L. Breiman, J. Friedman, R. Olshen, and J. Stone, *Classification and Regression Tree*. Chapman and Hall/CRC, 1984.

[50] A. Mortensen, R. Moskowitz, and T. Reddy, "Ddos open threat signaling requirements," Internet-Draft Working Draft http://bit.ly/2536nXl, October 2015.

[51] R. Dobbins, S. Fouant, D. Migault, R. Moskowitz, N. Teague, and L. Xia, "Use cases for DDoS open threat signaling," Internet-Draft http://bit.ly/24TyOn5, 2015.

[52] M. S. Kang, V. D. Gligor, and V. Sekar, "Spiffy: Inducing cost-detectability tradeoffs for persistent link-flooding attacks," 2016.

[53] J. Mirkovic and P. Reiher, "A taxonomy of ddos attack and ddos defense mechanisms," *ACM SIGCOMM CCR*, vol. 34, no. 2, pp. 39–53, 2004.

[54] G. Fedynyshyn, M. C. Chuah, and G. Tan, "Detection and classification of different botnet c&c channels," in *Proc. of ATC*, 2011.

[55] L. Bilge, D. Balzarotti, W. Robertson, E. Kirda, and C. Kruegel, "Disclosure: detecting botnet command and control servers through large-scale netflow analysis," in *Proc. of ACM CCS*, 2012.

[56] Y. Nadji, M. Antonakakis, R. Perdisci, D. Dagon, and W. Lee, "Beheading hydras: performing effective botnet takedowns," in *Proc. of ACM CCS*, 2013.

[57] C. Kanich, N. Weaver, D. McCoy, T. Halvorson, C. Kreibich, K. Levchenko, V. Paxson, G. M. Voelker, and S. Savage, "Show me the money: Characterizing spam-advertised revenue." in *Proc. of USENIX Security*, 2011.

[58] C. Y. Cho, E. C. R. Shin, D. Song *et al.*, "Inference and analysis of formal models of botnet command and control protocols," in *Proc. of ACM CCS*, 2010.

[59] S. Nagaraja, P. Mittal, C.-Y. Hong, M. Caesar, and N. Borisov, "Botgrep: Finding p2p bots with structured graph analysis." in *Proc. of USENIX Security*, 2010.

[60] P. M. Comparetti, G. Wondracek, C. Kruegel, and E. Kirda, "Prospex: Protocol specification extraction," in *Proc. of IEEE S&P*, 2009.

[61] W. Chang, A. Mohaisen, A. Wang, and S. Chen, "Measuring botnets in the wild: Some new trends," in *Proc. of ACM ASIACCS*, 2015.

[62] L. Wang, A. Liu, and S. Jajodia, "Using attack graphs for correlating, hypothesizing, and predicting intrusion alerts," *Computer Communications*, vol. 29, no. 15, pp. 2917–2933, 2006.