An Enhancing Framework for Botnet Detection Using Generative Adversarial Networks

Chuanlong Yin
State Key Laboratory of Mathematical Engineering and Advanced Computing
Zhengzhou, China
e-mail: dragonyincl@163.com

Yuefei Zhu (corresponding author)
State Key Laboratory of Mathematical Engineering and Advanced Computing
Zhengzhou, China
e-mail: zyf0136@sina.com

Shengli Liu
State Key Laboratory of Mathematical Engineering and Advanced Computing
Zhengzhou, China
e-mail: 475737@qq.com

Jinlong Fei
State Key Laboratory of Mathematical Engineering and Advanced Computing
Zhengzhou, China
e-mail: feijinlong@126.com

Hetong Zhang
State Key Laboratory of Mathematical Engineering and Advanced Computing
Zhengzhou, China
e-mail: hetongzhang@live.cn

***Abstract*—The botnet, as one of the most formidable threats to cyber security, is often used to launch large-scale attack sabotage. How to accurately identify the botnet, especially to improve the performance of the detection model, is a key technical issue. In this paper, we propose a framework based on generative adversarial networks to augment botnet detection models (Bot-GAN). Moreover, we explore the performance of the proposed framework based on flows. The experimental results show that Bot-GAN is suitable for augmenting the original detection model. Compared with the original detection model, the proposed approach improves the detection performance, and decreases the false positive rate, which provides an effective method for improving the detection performance. In addition, it also retains the primary characteristics of the original detection model, which does not care about the network payload information, and has the ability to detect novel botnets and others using encryption or proprietary protocols.**

Keywords-generative adversarial networks; botnet; botnet detection; deep learning; anomaly detection

I. INTRODUCTION

Botnet [1-3] is a network of computers (bots) infected with the same malware, and is under the control of hackers. It not only integrates traditional viruses and Trojans, performing system resident, information theft, remote control, and so on, but also has the ability of infection and propagation similar to network worms. Botnets are often used to launch large-scale cyber-attack sabotage. According to recent researches, almost all current distributed denial of service attacks come from botnets, 80% to 95% of spam is botnet-initiated. Moreover, click fraud, sensitive information theft, encryption extortion, and so on, also mainly resort to botnets for financial gain illegally [4-5]. Botnets are still the most popular tools for hackers on the Internet.

The network-based detection methods [6-8] mainly study the anomaly characteristics of botnets based on network flows. Through the induction, extraction, and selection of features based on statistical analysis, machine learning, data mining and other methods, the detection model is established, trained and optimized successively, and then is used for detectingbotnets. The network-based detection methods usually have relatively low detection accuracy, but they can detect novel botnets and others using encryption or proprietary protocols. Moreover, they take the passive monitoring and analysis of network flows as the basis of judgment, which are independent of the specific payload information and do not involve privacy issues. Therefore, the network-based detection methods become the mainstream approach in the field of botnet detection.

However, such methods suffer from two main issues. On the one hand, most existing network-based methods for botnet

978-1-5386-6987-7/18/$31.00 ©2018 IEEE

detection are limited to the packet inspection level, and most methods also mainly focus on partial characteristics of network flows, which cannot fully characterize the abnormal behaviors of botnets; On the other hand, botnets keep pace with the times and take advantage of advanced ideas and technologies to escape from detection. As technologies continue to evolve, botnets become increasingly sophisticated and intelligent, to a certain extent, exhibiting complexity and confrontation, which makes the security situation still grim.

Hence, it is still a difficult and hot topic in academia and industry to detect the botnet efficiently and accurately. How to identify botnets, especially to improve the performance of the detection model, is a key technical issue.

In this paper, we aim to propose a deep learning framework based on generative adversarial networks [9] (GAN) to improve the performance of the original botnet detection model, and mainly study the following work.

1) First, according to 5-tuple <source IP, destination IP, source port, destination port>, the network packets are merged into flows. Furthermore, we describe abnormal behaviors, similarity, and other characteristics of various types of botnets, and related features are extracted and selected correspondingly.
2) Second, the idea of generative adversarial networks is introduced into the field of botnet detection, and a botnet detection framework based on generative adversarial networks (Bot-GAN) is proposed for enhancing detection performance of the original model. The generative model in the framework continuously generates 'fake' samples to assist the original detection model to improve the performance.
3) Third, we present the detection performance, e.g., precision, recall, f1-score, and confusion matrix of the enhanced model via Bot-GAN. Under the same conditions, the performance of the enhanced model is compared with that of the original detection model. In addition, the enhanced model is tested to detect unknown botnets and others using encryption or proprietary protocols.

The experimental results show that the proposed framework is suitable for augmenting the original detection model. It improves the detection performance, decreases the false positive rate, and provides an effective approach for improving the botnet detection performance. Moreover, the proposed approach also retains characteristics of the original detection model, which does not care about the internal payload, and has the ability to detect unknown botnets and others using encryption or proprietary protocols.

II. Relevant Work

In the past decades, researchers introduced a variety of traditional machine learning and data mining methods for botnet detection and made great progress. In 2014, Shree Gary et al. [10] proposed a P2P botnet detection method based on network behavior analysis and used a decision tree to establish a botnet detection model, which can effectively extract malicious traffic from the normal network traffic. In 2017, Wang et al. [11] proposed a botnet detection method based on multidimensional permutation entropy and clustering variance. The authors calculated the complexity of time series of network traffic using multidimensional permutation entropy, and then detected the self-similarity based on cluster variance.

As deep learning [12-14] has developed rapidly, the related theoretical achievements and practical results are endless, especially in the field of speech recognition and image recognition, and have achieved remarkable results. The developments of theory and technology about deep learning have opened up a new era in artificial intelligence and also provided a completely new idea for botnet detection technology.

In [15], deep belief network (DBN) was introduced into the field of intrusion detection, and a deep belief network for intrusion detection system was proposed, which effectively improved the classification accuracy. In [16], authors applied the deep learning approach to software defined network (SDN) in 2016, and proposed a DDoS detection system based on sparse auto-encoder learning. The above literatures treat the deep learning method as a generative model to reduce features in the pre-training stage, and then utilize the traditional supervised learning to identify and classify the network flows.

In [17], Sneha Kudugunta et al. proposed a deep neural network (DNN) architecture to exploit both content and metadata to detect bots. In [18], a novel LSTM based algorithm was presented to handle the multiclass imbalance problem in botnet detection. In [19], the recurrent neural networks (RNN) algorithm was introduced to detect botnets by modeling network traffic as a sequence of states that change over time.

The generative adversarial networks (GAN) proposed by Goodfellow [9] in 2014 is a deep learning model and provides a new framework for generative models. It is one of the most promising methods for unsupervised learning in recent years. The framework learns from each other through the generator and discriminator to generate the simulated data. The generator captures the distribution of real data, while the discriminator estimates the probability that a sample come from training set.

In [20], the concept of GAN was leveraged to construct a deep learning method based domain generation algorithm (DGA) to intentionally bypass a deep learning based detector. At present, most researchers mainly regard GAN as a generative model, and mainly apply it to image super-resolution [21], text to image synthesis [22], image-to-image translation [23], and speech enhancement [24]. Unlike other variants of GAN, we pay more attention to the discriminator rather than the generator in this paper. Hence, inspired by GAN, we propose an enhanced framework for botnet detection based on generative adversarial networks (Bot-GAN), which continuously generates 'fake' samples by the generator and expands the number of labeled samples to assist the original model for botnet detection and classification. The experimental results show that the proposed framework improves the performance and generalization of the original model.

III. Proposed Methodologies

A. Generative Adversarial Networks

The core idea of GAN comes from game theory, which includes a generator and a discriminator, the generator's goal is to try to generate a real data distribution to deceive the discriminator, and the discriminator as much as possible correctly determines whether the input data comes from real data or from the generator. In order to achieve the final victory, these two game participants need to constantly optimize themselves to improve their ability to reach Nash equilibrium. As mentioned above, the training process is summarized as follows:

$$\min_G \max_D V(D,G) = E_{x\sim P_{data}}[\log D(x)] + E_{z\sim P_z}(z)\left[\log(1-D(G(z)))\right] \quad (1)$$

Theoretically, any differentiable function can be used as the generator and discriminator. The letters D and G represent the discriminator and the generator, respectively. Their inputs are the real data x and the random variable z. $G(z)$ represents the samples generated by G that captures the real data distribution p_{data}. If the discriminator's input comes from real data, it is labeled as 'real'. If the input sample is from $G(z)$, it is labeled as 'fake'. The goal of D here is to achieve a binary classification of data sources: real (from a distribution of real data x) or fake (from generated data $G(z)$), The objective is to make the performance $D(G(z))$ of the generate data $G(z)$ on D consistent with the performance $D(x)$ of the real data x on D. In the mutually antagonistic and iterative optimization processes, the performance of D and G continues to be improved together. When the discriminative ability of D is raised to a certain extent and the data source cannot be correctly identified, the generator G can be considered as the final model which has learned the distribution of real data.

B. Proposed Enhanced Framework

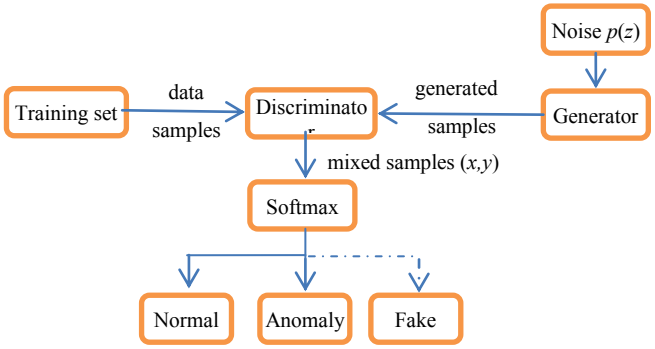

Figure 1. The Framework of Bot-GAN.

The discriminator in a standard GAN is essentially a binary classifier that classifies the samples into real or fake categories. In fact, real samples can be further categorized into normal traffic and abnormal traffic for a botnet detector. Hence, we replaced the discriminator with a botnet detector, and transformed its binary output (normal, anomaly) into a triple output (normal, anomaly, fake) using the softmax function. Then a framework based on generative adversarial networks for botnet detection (Bot-GAN) is proposed shown in Figure 1. Obviously, the generator in the framework can continuously generate 'fake' samples, which add a class with 'fake' label to the original training set. Meanwhile, the binary outputs (normal and anomaly) of the original botnet detection model are extended to triple outputs (normal, anomaly, and fake).

IV. Experiments

A. Dataset

It is well known that the performance of botnet detection methods in practice is highly dependent on the dataset used to evaluate the detection performance. Taking into account the diversity, universality, privacy protection and other issues, we used the ISCX botnet dataset [25] created by the University of New Brunswick as a benchmark dataset in this paper. The dataset with real labels includes 16 different types of botnets traffic and overcomes the problems of traditional botnet datasets.

The training set includes 4 different types (Neris, Rbot, Virut and NSIS) of botnets traffic from ISOT dataset [26], some normal traffic from ISCX 2012 intrusion detection dataset [27] and malware capture projects [28]; the test set contains 25% from ISOT dataset, some normal traffic and botnets traffic from ISCX 2012 intrusion detection, and 9 different types (Neris, Rbot, Virut, NSIS, Menti, Sogou and Murlo botnets) of botnets traffic from the malware capture project.

Table 1 describes the different types of botnets in the training and test sets in detail. There are more types of botnets in the test set than that in the training set, which ensures that the test set can evaluate whether the detection model has the ability to detect novel botnets.

TABLE I. ISCX Botnet Dataset

Botnets	Type	Training set	Test set
Neris	IRC	√	√
Rbot	IRC	√	√
Menti	IRC	×	√
Sogou	HTTP	×	√
Murlo	IRC	×	√
Virut	HTTP	√	√
NSIS	P2P	√	√
SMTP Spam	P2P	√	√
Zeus	P2P	√	√
UDP Storm	P2P	×	√
Tbot	IRC	×	√
Zero Access	P2P	×	√
Weasel	P2P	×	√
Smoke Bot	P2P	×	√
Zeus control (C & C)	P2P	√	√
ISCX IRC bot	P2P	×	√

B. Features Extraction and Selection

Most existing methods for botnet detection based on network focus on partial attributes of network flows, which cannot fully characterize abnormal behaviors, and cannot fully describe the attack process of botnets. In practice, it leads to a low detection performance. With the continuous development of technology, the network system is more and more complicated, and the network traffic shows some instability and complexity. Therefore, a network system or a host system cannot be described solely from the point of the network packets or the local attribute of the network traffic. Hence, we need to fully analyze the attributes of network flows in combination with other features.

Based on the existing researches [25-34], we deeply studied the characteristics of communication, similarity, and abnormal behaviors based on network flows, combed and combined some repetitive features, and further optimized the features extraction and selection. As shown in Table 2, 16 typical features are selected in our research.

TABLE II. SUMMARY OF FEATURES BASED ON NETWORK FLOWS

No.	Features	Description
1	Protocol	Protocol
2	PX	total number of packets exchanged
3	NNP	number of null packets exchanged
4	PSP	percentage of small packets exchanged
5	IOPR	ratio between the number of incoming packets over the number of outgoing packets
6	Reconnect	number of reconnects
7	Duration	flow duration
8	FPS	length of the first packet
9	TBT	total number of bytes
10	APL	average payload packet length
11	DPL	total number of packets with the same length over the total number of packets
12	PV	Standard deviation of payload packet length
13	BS	average bits-per-second
14	PS	average packets-per-second in a time window
15	AIT	average inter arrival time of packets
16	PPS	average packets-per-second

C. Preprocessing

1) Flow Processing. First, we need to reassemble network packets into flows. In accordance with 5-tuple <source IP, destination IP, source port, destination port, protocol >, the packets are re-combined into flows, and then each flow is labeled as normal or anomaly. Finally, 16 features based on flows are extracted. After processing, the training set consists of 491,381 flows, of which 192,112 (39.10%) flows are malicious, and contains 7 types of botnets; the test set consists of 348452 flows, of which 169988 (48.78%) flows are malicious, and contains 16 types of botnets.

2) Numeralization. Each flow consists of 16 features, including 15 numeric features and 1 nonnumeric feature, which need to be converted into numeric form. There are 107 kinds of values for the feature 'protocol', which has to be encoded into 107-dimensional vectors. 16 features are mapped into 122-dimensional vectors.

3) Standardization. Using standard deviations (Equation 2), we scaled the value of each feature between [-1, 1]:

$$x_i = \frac{x_i - x_{mean}}{x_{std}} \quad (2)$$

where x_i is the value of each feature, x_{mean} is the average value of each feature, and x_{std} is the standard deviation of each feature.

D. Evaluation Criteria

In order to verify the performance of botnet detection models, we evaluate the performance in terms of accuracy, precision, recall, false positive rate (FPR), f1-score, and confusion matrix. **TP** indicates the number of illegal flows correctly detected, **TN** indicates the number of legitimate flows correctly identified, **FN** indicates the number of illegal flows that are incorrectly detected as legal flows, and **FP** indicates the legal flows that are incorrectly detected as illegal flows.

In the field of machine learning and artificial intelligence, confusion matrix as a visualization tool, is very suitable for supervised learning. As shown in Table 3, each row of the confusion matrix represents the real category of flows, and the total number of each row represents the total number of flows as the category; each column of the confusion matrix represents a predicted class, and the total number of each column represents the number of flows predicted as the category.

TABLE III. CONFUSION MATRIX

Actual Class \ Predicted Class	anomaly	normal
anomaly	TP	FN
normal	FP	TN

The accuracy, precision, recall, false positive rate, and f1-score are respectively defined in equations (3) to (7).

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (4)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (5)$$

$$\text{F1 score} = \frac{2\times Recall \times Precision}{Recall + Precision} \quad (6)$$

$$\text{False positive rate} = \frac{FP}{TN+FP} \quad (7)$$

Therefore, a good botnet detection model should have a higher accuracy, precision, recall, f1-score, and a lower false positive rate.

E. Bot-GAN Setup

In the original GAN theory, *G* and *D* are not required to be neural networks. In practice, however, deep learning models are generally used as *G* and *D*.

In our research, we select a 3-layer LSTM network as G, which includes an input layer, a hidden layer and an output

layer. There are 120 neurons in the input layer, and the number of hidden layer nodes is set to 80. There are 122 output nodes, the same as the number of processed features mentioned in preprocessing.

A 4-layer neural networks structure is adopted as the detection model, including an input layer, two hidden layers and an output layer. There are 122 neurons in the input layer, which is the same as the number of input features. There are 80 and 20 nodes in the first and second hidden layers, respectively. The number of the output layer is 3, which is the same as the number of classes.

V. Results and Discussion

The entire experiments were executed on a personal computer by Intel Core i7-5500U CPU @ 2.40 GHz, 8 GB memory. In order to ensure the accuracy and objectivity of experiments, the structure and parameters of the detection model (the original model) in the comparative experiment is consistent with that of the 4-layer neural networks selected in the Bot-GAN framework, except that it has 2 nodes in the output layer (normal or anomaly).

Based on the ISCX botnet dataset, we iterated 100 times on the training set to get the best model and observed the accuracy, precision, recall, f1-score and false alarm rate of the original model on the test set. As shown in Figure 2, the results we obtained in the experiments are equivalent to that in [25] based on the same dataset.

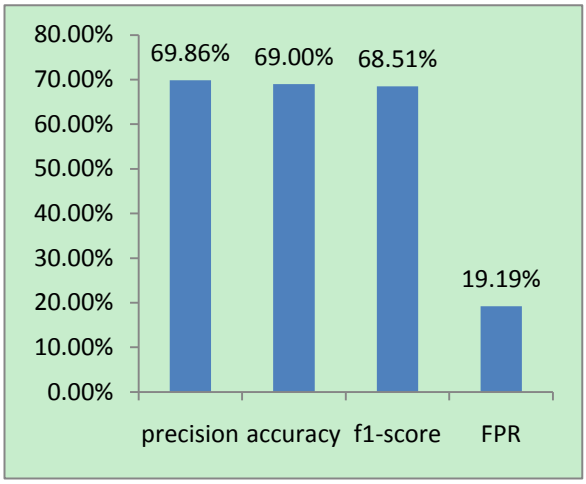

Figure 2. The detection performance of the original model on the test set.

In order to ensure the adequacy of experiments, and objectively evaluate the performance of the proposed framework to enhance the original model. Similarly, based on the ISCX botnet dataset, we separately mixed 100, 500, 1000, 2000, 5000 and 8000 different numbers of generated samples with the training set, on which we trained the detection model 100 times, and observed the accuracy, precision, and other performance indicators of the enhanced model via Bot-GAN on the test set.

In terms of precision, the precision of the enhanced model via GAN is higher than that of the original model on the test set after mixed with different numbers of generated samples. In particular, the enhanced model obtains the maximum value on the test set when mixing 1000 generated samples. Of course, with the increase of generated samples, the detection precision of the enhanced model is gradually reduced. When the mixed number reaches 8000, the result is equivalent to that of the original model.

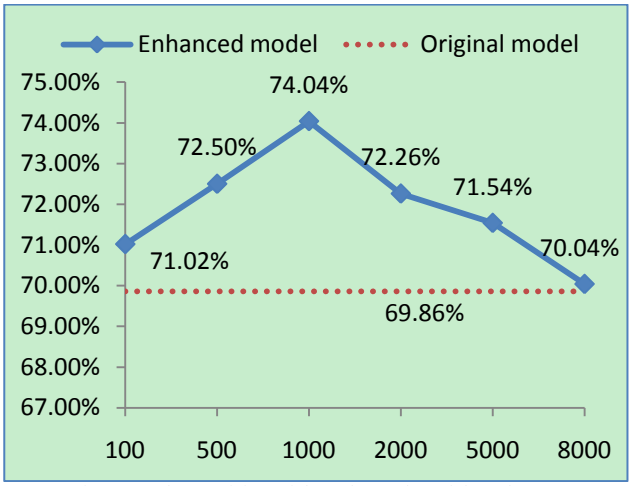

Figure 3. The precision of the enhanced model on the test set.

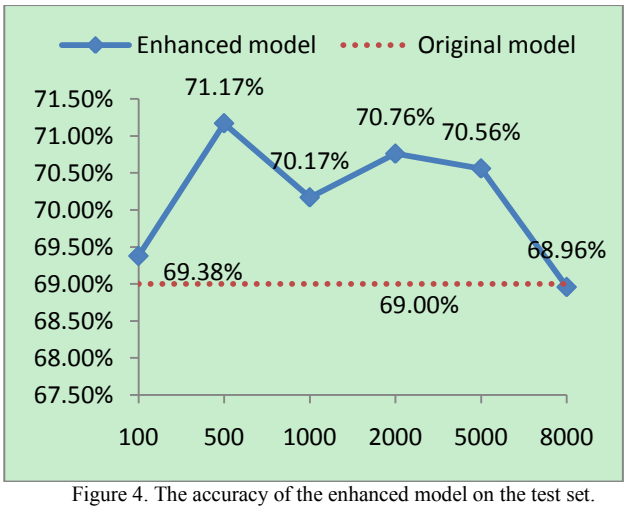

Figure 4. The accuracy of the enhanced model on the test set.

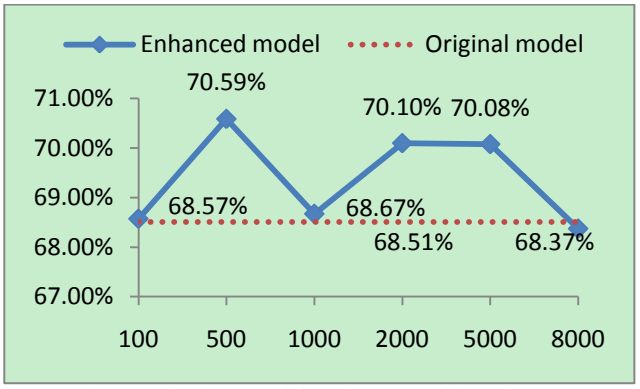

Figure 5. The f1-score of the enhanced model on the test set.

In terms of recall and f1-socre, when mixing 100, 500, 1000, 2000, and 5000 different numbers of generated samples,

the accuracy and f1-score of the enhanced model are superior to the original model on the test set shown in Figure 4 and 5, respectively. In particular, the enhanced model obtains the maximum values of recall and f1-score on the test set when the mixed number is 500. Similarly, as the generated samples increase, the performance of the enhanced model decreases gradually with the recall and f1-score. When mixing 8000 generated samples, the result is inferior to that of the original model.

TABLE IV. CONFUSION MATRIX OF THE ORIGINAL CLASSIFIER ON KDDTEST+

Actual Class \ Predicted Class	anomaly	normal
anomaly	96231	73757
normal	34254	144210

TABLE V. CONFUSION MATRIX OF THE ORIGINAL CLASSIFIER (MIXED NUM.=500) ON KDDTEST+

Actual Class \ Predicted Class	anomaly	normal
anomaly	97366	72622
normal	27828	150636

Tables 4 and 5 show the confusion matrix of the original detection model and the enhanced model (mixed number is 500) via the Bot-Gan framework on the test set, respectively. At this moment, the false positive rate of the enhanced model dropped from 19.19% to 15.59% on the test set.

In order to assess the ability of the detection model to detect novel botnets, more types of botnets are guaranteed in the test set than the training set. The experimental results show that the enhanced model not only can detect IRC botnets and P2P botnets, but also can detect some types of botnets that appear in the test set and not in the training set, that is, the enhanced model, like the original model, has the ability to detect some novel botnets and others utilizing encryption or proprietary protocols.

In conclusion, the enhanced botnet detection model based on the generative adversarial networks improves the precision, accuracy, and other performance indicators, reduces the false positive rate and enhances the botnet detection performance and generalization of the original model. In addition, the enhanced model still retains the primary characteristics of the original model, and has the ability to detect the novel botnets.

VI. CONCLUSION AND FUTURE WORK

The Bot-GAN framework proposed in this paper can continuously generate 'fake' samples by the generator, which can expand the labeled data by proper amount. It can effectively improve the precision, accuracy and other performance indicators, and reduce the false positive rate of the original model. It provides a universal framework for enhancing the performance of botnet detection model. In future work, we will focus on the enhanced ability of the framework for different types of detection models.

ACKNOWLEDGMENTS

This work was supported in part by the National Key Research and Development Program of China (NO.2016YFB0801601 & NO.2016YFB0801505).

REFERENCES

[1] Feily, Maryam, Alireza Shahrestani, and Sureswaran Ramadass. "A survey of botnet and botnet detection." *Emerging Security Information, Systems and Technologies, 2009. SECURWARE'09. Third International Conference on*. IEEE, 2009.

[2] Zhu, Zhaosheng, et al. "Botnet research survey." *Computer Software and* Applications*, 2008. COMPSAC'08. 32nd Annual IEEE International*. IEEE, 2008.

[3] Bailey, Michael, et al. "A survey of botnet technology and defenses." *Conference For Homeland Security, 2009. CATCH'09. Cybersecurity Applications & Technology*. IEEE, 2009.

[4] John, John P., et al. "Studying Spamming Botnets Using Botlab." *NSDI*. Vol. 9. 2009.

[5] Hoque, Nazrul, Dhruba K. Bhattacharyya, and Jugal K. Kalita. "Botnet in DDoS attacks: trends and challenges." *IEEE Communications Surveys & Tutorials* 17.4 (2015): 2242-2270.

[6] Feily, Maryam, Alireza Shahrestani, and Sureswaran Ramadass. "A survey of botnet and botnet detection." *Emerging Security Information, Systems and Technologies, 2009. SECURWARE'09. Third International Conference on*. IEEE, 2009.

[7] Qiu, Zhicong, David J. Miller, and George Kesidis. "Flow based botnet detection through semi-supervised active learning." *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017.

[8] Lashkari, Arash Habibi, et al. "A Survey Leading to a New Evaluation Framework for Network-based Botnet Detection." *Proceedings of the 2017 the 7th International Conference on Communication and Network Security*. ACM, 2017.

[9] Goodfellow, Ian, et al. "Generative adversarial nets." *Advances in neural information processing systems*. 2014.

[10] Garg, Shree, Anil K. Sarje, and Sateesh Kumar Peddoju. "Improved detection of P2P botnets through network behavior analysis." *International Conference on Security in Computer Networks and Distributed Systems*. Springer, Berlin, Heidelberg, 2014.

[11] Wang, Jiajia, and Yu Chen. "Botnet Detection Method Based on Permutation Entropy and Clustering Variance." *DEStech Transactions on Engineering and Technology Research* ismii (2017).

[12] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." *nature* 521.7553 (2015): 436.

[13] Schmidhuber, Jürgen. "Deep learning in neural networks: An overview." *Neural networks* 61 (2015): 85-117.

[14] Goodfellow, Ian, et al. *Deep learning*. Vol. 1. Cambridge: MIT press, 2016.

[15] Gao, Ni, et al. "An intrusion detection model based on deep belief networks." *Advanced Cloud and Big Data (CBD), 2014 Second International Conference on*. IEEE, 2014.

[16] Niyaz, Quamar, Weiqing Sun, and Ahmad Y. Javaid. "A deep learning based DDoS detection system in software-defined networking (SDN)." *arXiv preprint arXiv:1611.07400* (2016).

[17] Kudugunta, Sneha, and Emilio Ferrara. "Deep Neural Networks for Bot Detection." *arXiv preprint arXiv*:1802.04289 (2018).

[18] Tran, Duc, et al. "A LSTM based framework for handling multiclass imbalance in DGA botnet detection." *Neurocomputing* 275 (2018): 2401-2413.

[19] Torres, Pablo, et al. "An analysis of Recurrent Neural Networks for Botnet detection behavior." *Biennial Congress of Argentina (ARGENCON), 2016 IEEE*. IEEE, 2016.

[20] Anderson, Hyrum S., Jonathan Woodbridge, and Bobby Filar. "DeepDGA: Adversarially-tuned domain generation and

detection." *Proceedings of the 2016 ACM Workshop on Artificial Intelligence and Security*. ACM, 2016.

[21] Ledig, Christian, et al. "Photo-realistic single image super-resolution using a generative adversarial network." *arXiv preprint* (2016).

[22] Reed, Scott, et al. "Generative adversarial text to image synthesis." *arXiv preprint arXiv:1605.05396* (2016).

[23] Isola, Phillip, et al. "Image-to-image translation with conditional adversarial networks." *arXiv preprint* (2017).

[24] Pascual, Santiago, Antonio Bonafonte, and Joan Serra. "SEGAN: Speech enhancement generative adversarial network." *arXiv preprint arXiv:1703.09452* (2017).

[25] Beigi, Elaheh Biglar, et al. "Towards effective feature selection in machine learning-based botnet detection approaches." *Communications and Network Security (CNS), 2014 IEEE Conference on*. IEEE, 2014.

[26] Zhao, David, et al. "Botnet detection based on traffic behavior analysis and flow intervals." *Computers & Security* 39 (2013): 2-16.

[27] Shiravi, Ali, et al. "Toward developing a systematic approach to generate benchmark datasets for intrusion detection." *computers & security* 31.3 (2012): 357-374.

[28] García, S. "Malware capture facility project." *cvut* (2013).

[29] Saad, Sherif, et al. "Detecting P2P botnets through network behavior analysis and machine learning." *Privacy, Security and Trust (PST), 2011 Ninth Annual International Conference on*. IEEE, 2011.

[30] Yu, Xiaocong, et al. "Data-adaptive clustering analysis for online botnet detection." *Computational Science and Optimization (CSO), 2010 Third International Joint Conference on*. Vol. 1. IEEE, 2010.

[31] Liao, Wen-Hwa, and Chia-Ching Chang. "Peer to peer botnet detection using data mining scheme." *Internet Technology and Applications, 2010 International Conference on*. IEEE, 2010.

[32] Livadas, Carl, et al. "Usilng machine learning technliques to identify botnet traffic." *Local Computer Networks, Proceedings 2006 31st IEEE Conference on*. IEEE, 2006.

[33] Strayer, W. Timothy, et al. "Botnet detection based on network behavior." *Botnet Detection*. Springer, Boston, MA, 2008. 1-24.

[34] Zhao, David, et al. "Peer to peer botnet detection based on flow intervals." *IFIP International Information Security Conference*. Springer, Berlin, Heidelberg, 2012.