Anomaly Detection in DNS Query Logs using Improved Binary Black Hole Optimization Algorithm

[1]S.S.Suganya, [2]Dr.V.Kathiresan

[1] Department of Computer Science, Dr.SNSRajalakshmi college of Arts & Science, Coimbatore – 641 049, Tamil Nadu, India.

[2] Department of Computer Applications(PG), Dr.SNSRajalakshmi college of Arts & Science, Coimbatore – 641 049, Tamil Nadu, India.

[1]suganya.annur@gmail.com

[2]vkathirmca@gmail.com

Abstract Domain Name System (DNS) log information supplies a unique perspective on domain names usage by both legitimate users and anomaly users. More than analyzing DNS queries in traditional manner, this research work aims in design and development of binding approach based on Improved Binary Black Hole Optimization Algorithm IBBHOA for feature selection in SVM classifier. At first unremitting black hole optimization algorithm is portrayed. Next an improved binary black hole optimization algorithm is presented. After that binding approach based on IBBHOA for feature selection in SVM Classifier is carried out. SVM classifier is chosen for performing the classification task for characterizing DNS lookup behaviors by means of log-mining. DNS query logs are obtained from the dataset from various sources. Feature selection is performed and then the SVM classifier is used to classify anomaly behaviors, DNS failed requests, time taken for feature selection and time taken for classification are the performance metrics chosen for comparison.

Keywords: DNS, query logs, support vector machine, feature selection, anomaly behavior.

1. Introduction

DNS (Domain Name System) is a fundamental protocol for the other applications, such as web or mail, in the internet. For example, URL, which indicates the location of a web resource, contains a domain name string. When a host accesses to a web resource, the host has to resolve the IP address of the name. An e-mail address is composed of a user name and a domain name, which is used by the mail exchanger. Therefore, DNS queries are closely related to hosts' behavior, such as web browsing or sending e-mails. Since the volume of traffic is increasing year by year, it is becoming difficult to monitor individual hosts. DNS is useful for monitoring host anomalies in a network management. Alternative methods that monitor only DNS traffic rather than whole traffic have been proposed. Jung et al. proposed that DNS traffic be available to observe anomalies [1].

On the other hand, a DNS server itself is also involved in attacks, such as a DDoS (Distributed Denial of Service) attack and a cache poisoning[2] attack. DNS is a protocol on UDP, which the source IP address can be spoofed by an attacker so that the attacker can exploit DNS. For example, a DNS server can be a reflector of a DRDoS (Distributed Reflected DoS) attack queried spoofed packets with the targeted host. In this research work, binding approach based on IBBHOA for feature selection in SVM classifier is used to detect anomalies, which include both internal/external and clients/servers, monitoring DNS servers in a network. There are certain number of previous research works carried out that detect host anomalies monitoring DNS servers; however, these proposals cannot work well in a network where many hosts am observed or for long-term tracing. Since they record data for every host, the amount of computational resource required depends on the number of hosts. There are some other researches that target DNS servers in a large network however they do not target individual hosts but domain names. They analyze statistics of queries, and find out abnormal domain names controlled by an attacker. For example, they detect domain names used by a bot network, which is a network of compromised PCs, and employ DNS resource records to control them. Hence there is vast scope in this research area.

2. Literature Review

Passive Testing or offline analysis of network protocols has been reached its maturity since its first application in network fault detection by Boloutas et al. [2], the main idea in PT is to model the protocol as a finite state machine (FSM), extended finite state machine (EFSM) [3], and/or communicating finite state machines (CFSM) [4], [5], and then test network traces against the model to check conformance/performance.

There are several related works that detect anomalies by monitoring a DNS server. White et al. showed that DNS is available for monitoring anomalies in a network [6]. They relied on correlation between DNS's and the other protocol's traffic in an enterprise network, and detected the propagation of worms. However, it needs the other protocols' traffic also, and cannot detect anomalies without other traffic. Musashi et al. and other successors proposed host anoma-lies detection monitoring only DNS servers [7], [8].

Ren et al. presented Flying Term, a new perceptually motivated visual metaphor for visualizing the dynamic nature of DNS queries [9]. Hadi et al. offered a comprehensive review of network security visualization and provided a taxonomy in the form of five use-case classes encompassing nearly all recent works in this area [10]. Schonewille and Helmond's research was a first glance at the usability of DNS traffic and logs for detection of this malicious network activity. It is possible to detect the bots through the information od DNS gathered from the network by placing counters and triggers on specific events in the data analysis [11]. David and Paul considered three classes of DNS traffic: canonical, overload and unwanted, and showed preliminary results on how DNS analysis could be coupled with general network traffic monitoring to provide a useful perspective for network management and operations [12]. Kirkpatrick et al. introduced a method for clustering misconfigured DNS sources [13]. Using machine learning methods, they analyzed 24 h of DNS requests that were collected on the A-root DNS server. Their research provided preliminary results that were validated via discussion with DNS system operators. Shan et al. proposed an interactive visual analysis system for the DNS log files to intuitively detect the anomalies in DNS query logs [14]. Albrecht-Buehler used motion to visualize trends among text–theme relationships and allowed user interaction of the temporal controls and theme relations [15]. Brandes et al. used animation to illustrate the dynamics of international political and military conflicts [16]. In the research work of Pieter's an approach namely visual analytics is used on a huge set of DNS packet captures into ways that authoritative name servers were abused for denial of service attacks [17]. Several tools were developed to identify patterns in DNS queries and responses.Visualization analysis tool was presented by Yu for analyzing, catching and acknowledging to the Distributed Denial of Service attack termed the Domain Name Service (DNS) amplification attack [18]. In Born's study both quantitative analysis and visual aids were provided that allowed the user to make determinations about the legitimacy of the DNS traffic [19].

3. Proposed Work

The proposed work introduces binding approach based on IBBHOA for feature selection in SVM classifier. At first unremitting black hole optimization algorithm is portrayed. Next an improved binary black hole optimization algorithm is presented. After that binding approach based on IBBHOA for feature selection in SVM Classifier is carried out. SVM classifier is chosen for performing the classification task for characterizing DNS lookup behaviors by means oflog-mining. DNS query logs are obtained from the dataset from various sources. Feature selection is performed and then the SVM classifier is used to classify anomaly behaviors, DNS failed requests, time taken for feature selection and time taken for classification are the performance metrics chosen for comparison.

3.1. Unremitting Black Hole Optimization Algorithm (UBHOA)

The black hole optimization algorithm is a robust stochastic optimization technique based on simulation of the behavior of black hole in external space. The below steps explain manner of simulating UBHA from black hole phenomenon:

Step 1: Outer space is full of known and unknown stars. In real space black hole is formed by collapsing individual stars so UBHOA begins with the population of stars that located arbitrarily in the explore space. In UBHOA each star has a fitness value, which is evaluated by a fitness function to be optimized. The best star that has the best fitness value is selected as the black hole.

Step 2: In the real space, a black hole is an object of extreme density with an intense gravitational attraction. This leads to a great amount of gravitational force pulling stars around it. UBHOA has followed the same behavior. By Eq. (1) all the stars began moving toward the black hole.

Step 3: The sphere shaped bound of a black hole in outer space is known as the event horizon. The event horizon radius is called as the Schwarzschild radius. The red circle in Fig. 1 shows the event horizon of black hole. In the real space the Schwarzschild radius is computed by Eq. (2) and in UBHOA is computed by Eq. (3).

Step 4: Because of extreme density and strong gravitational attraction of black hole when a star crosses the event horizon, it will be swallowed by the black hole and disappear. In the region of event horizon the escapee speed is tantamount to the speed of the light, so nothing can get away from within the event horizon. In UBHOA, the Euclidean distance between black hole and star is computed. If this distance is less than Schwarzschild radius, substitute it with a fresh star in the random location in the search space.

Step 5: In UBHOA if a star reaches a location with lower cost than the black hole, in that case theirs locations need to be altered.

$$X_i(t+1) = X_i(t) + rand \times (X_{BH} - X_i(t)) \; i = 1, 2 \ldots N \ldots (1)$$

$$R = 2GM / C^2 \ldots (2)$$

$$R = f_{BH} \; / \sum_{i=1}^{N} f_i \ldots (3)$$

where $X_i(t)$ and $X_i(t+1)$ signify the locations of the $i\,th$ star at *iterations* t and $(t+1)$, respectively. $Rand$ indicates uniform distribution with a range from 0 to 1. N denotes the number of stars. *XBH* points the location of the black hole in the exploration space. M , G , and C signify the mass of the black hole, the gravitational constant, and the speed of light respectively. f_i denotes the fitness value of the i th star and f_{BH} indicates the fitness value of the black hole.

The framework of the UBHOA method is presented in Algorithm 1.

Algorithm 1. The unremitting black hole algorithm

3.2. The Proposed Improved Binary Black Hole Optimization Algorithm (IBBHOA)

The UBHOA was originally developed for unremitting valued spaces. But there exist a number of discrete combinatorial optimization problems, such as feature selection, in which the values are not unremitting numbers but rather discrete binary integers. The unremitting black hole algorithm reason, we have introduced binary version of UBHOA and mentioned the same as IBBHOA. Binarization techniques can be categorized into two groups: Two steps binarization and unremitting-binary operator transformation. The proposed binarization technique belongs to the first group. In the first group without any modifications in the operators, only two steps are added after the unremitting iteration.

In solving feature selection problem the search space must be modeled as a d-dimensional Boolean web, where the i^{th} star moves around the d -dimensional space.

Since the problem is to select or not select of a given feature, the position of a star only takes the values 1 or 0. Therefore, a transfer function is needed to forces stars to move in a binary space. Transfer functions define the probability of changing position's elements from 0 to 1 and vice versa. In the proposed approach, Hyperbolic Tangent function is utilized to modify the position of stars as in the Eq. (4) and (5).

$$S(X_{id}(t+I)) = \text{abs}(\tanh(X_{id}(t+i))) \ldots (4)$$

$$X_{id}(t+i) = 1 \; If \; S(X_{id}(t+i)) > r \; and \; 0 \; otherwise \quad \ldots (5)$$

Where rand is a uniform random number between 0 and 1. In Eq. (5), instead of rand threshold 0.6 can also be considered. In IBBHOA we only need to set number of stars. The proposed algorithm does not suffer from some of other optimization algorithms difficulties such as the slow convergence rate and adjusting several parameters. Compared with other optimization algorithms, IBBHOA is easier to implement, depend on a single parameter for configuring the model, requires much less memory, and converges more rapidly.

3.3. The proposed Binding approach based on IBBHOA for Feature Selection in SVM Classifier

At the beginning of IBBHOA, the primary population of the star's position is initialized randomly. Each star encodes a candidate feature subset based on a bit string. The length of the string is equivalent to the total number of features in the dataset of interest. In the binary encoding, a bit of one implies the feature is chosen and a bit of zero means that the feature is not chosen. Similar to other optimization algorithms, the fitness value of each star is calculated by using an evaluator.

In the part of evaluating fitness value of stars, when two founded stars have identical fitness value, the one with smaller number of features is chosen as the best star (black hole).The procedure stops once stopping criteria (maximum number of iterations) is met. The parameters for IBBHOA specify 25 iterations of population consisting of 10 stars. At the end of the IBBHOA wrapper based FS algorithm, the star with the best performance is selected. The position of this star gives the selected features. In order to avoid producing random results and provide an assurance for impartial comparison of the classification performances, assessing the efficiency of SVM classifier for selected features by optimization algorithms is executed 100 times. SVM is a supervised machine learning classifier which is applied for categorization. SVM finds the best possible surface to separate the positive samples from the negative samples. SVM is comparatively better than that of text classification when compared to Naive Bayes (NB) classifier and maximum entropy based classifiers.

The fundamental aim of SVM during the training process is to hit upon a maximum margin hyperplane to solve the feature review's classification task. There exist limitless possible boundaries in order to break up the two different classes. For choosing the best class, it is significant to prefer a decision boundary which contains a maximum margin between any points from both classes. The decision boundary with a maximum margin would be less likely to make prediction errors, which is close to the boundaries of one of the classes. In this part of research a simplified SVM that is capable enough to classify multi-class and performs dual roles. In the beginning, making a model for the training data set and then using that model to conclude facts of a testing data set.

The SVM procedure includes the following steps.

Step – 1: Transform data to the format of an SVM package
Step – 2: Conduct simple scaling on the data.
Step – 3: Consider the RBF kernel.
Step – 4: Features are selected based on IBBHOA to train the whole training dataset.
Step – 5: Test with the testing dataset.

After preprocessing, the above procedures are carried out for training the SVM. The basic form of features and its classification is illustrated in the following equation.

$$\phi = (Ds \times C_{s)} \rightarrow \{P, N\} \ldots (6)$$

where D_S is a set of documents and C_S is a set of categories.

If $\emptyset : (D_S \times C_S) = P$, then D_{Si} is a positive member of C_{Sj}

If $\emptyset : (D_S \times C_S) = N$, then D_{Si} is called a negative member of C_{Sj}.

The SSVM method gives a positive value (+1) in the most appropriate holding data points and a negative value (−1) in rest of the places. Furthermore, the non-linear mapping function, that maps the training data can be defined as follows.

$$\emptyset: R^N \rightarrow R^F \ldots (7)$$

where R^N is a non-linear mapping that represents training data for feature space R^F. Hence, there is a need for performing optimization in order to segregate the dataset.

The kernel functions provide more decision functions when the data are nonlinearly separable. The kernel functions used the following polynomial function and Gaussian Radial-Basis Function (RBF). The RBF kernels of SVM are used in our system to build models. These models predict information for the testing data set. The representation points for each feature vector lay on a 1D plane and cannot be separated by a linear hyperplane. Therefore, the system will first use a kernel function that maps the points into feature space and then separates the points by hyperplane. The kernel function that will do the job is $k(xi, xj) = \emptyset (xi) \times \emptyset (xj)$. In addition, the kernel polynomial function maps the feature space points into 2D by multiplying the points to the power of two.

4. Results and Discussions

DNS traffic usually makes use of User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) on port 53 in order to perform the communication task [20]. Almost all the DNS communication are carried out through UDP, which is the default protocol used by resolvers, i.e. few applications those have communication with DNS servers for other applications when they need to resolve a DNS query. TCP was formerly has been made use for zone transfers, other than RFC 1123 [21] expanded the use of TCP as a backup communication protocol when the answer needs to be larger than 512 octets. With this connection, the first UDP DNS response has only fractional answers. The truncation bit is set with the intention that the resolver possibly replicates the query over TCP. On the other hand, RFC 2671, "EDNS0" [22], defined a new opcode field / pseudo resource record that allows UDP DNS traffic to be bigger than 512 octets. This is due to roughly all of today's DNS traffic uses UDP as its transport protocol. Sorting the obtained records by different criterion is used to detect unusual records or activities. At the same time as searching for records with low TTL values can generally be useful in detection of fast flux domains. The top 10 wireless client IP addresses and the details are shown in Table 1. Table 2 depicts the details which consists of queries per day, domain name, number of anomaly received requests and number of failed requests about the top 10 destination domains.

Algorithm 2. Pseudo code of IBBHOA for FS

Table I Top 10 wireless client IP addresses

Client S. No.	IP Address	Queries / Day	Number of Anomaly Requests Sent	Number of Failed Requests
1	139.59.17.152	2,41,056	40,338	10,650
2	59.99.150.200	2,28,159	19,436	8,080
3	223.179.220.177	1,14,311	12,719	4,412
4	118.151.209.5	95,981	9,736	3,417
5	203.90.4.145	84,452	7,642	3,062
6	103.234.190.183	74,191	5,618	3,391
7	122.176.20.6	73,445	3,749	3,234
8	150.107.25.248	65,049	5,085	2,545
9	112.133.201.135	62,237	4,896	2,947
10	103.1.115.219	52,617	3,827	1,865

Table II Top 10 destination domain names

Domain S. No.	Domain Name	Queries / Day	Number of Anomaly Received Requests	Number of Failed Requests
1	baidu.com	37,89,755	1,02,749	2,23,080
2	qq.com	26,61,189	91,836	1,71,689
3	akadns.net	23,87,316	84,782	1,53,196
4	apple.com	17,90,093	54,736	1,02,531
5	taobao.com	11,86,935	61,038	67,259
6	in-addr.arpa	10,89,644	64,954	67,533
7	google.com	9,81,895	92,743	67,197
8	weibo.com	7,98,301	90,664	47,710
9	sina.com.cn	7,85,648	86,498	49,648
10	360.cn	7,55,655	85,053	43,928

Table III IDetection Accuracy of Anomaly Requests Sentfrom Client Side

Client S.No.	PCA - SVM Classifier					IBBHOA - SVM				
	TP	TN	FP	FN	Accuracy	TP	TN	FP	FN	Accuracy
1	15301	14994	5045	4998	75.10	19473	13438	2739	4788	81.39
2	8521	5783	2221	2911	73.60	9523	6729	1239	1945	83.62
3	5857	3809	1721	1332	76.00	6192	4151	1394	982	81.32
4	4901	2592	1209	1034	76.96	4901	2892	1009	934	80.04
5	3671	2201	994	776	76.84	3872	2521	822	427	83.66
6	2193	2114	712	599	76.66	2236	2379	601	402	82.15
7	1351	1389	418	591	73.09	1623	1509	218	399	83.54
8	2125	1759	516	679	76.47	2388	1813	399	485	82.62
9	2217	1537	493	649	76.67	2488	1585	304	519	83.19
10	1548	1381	502	396	76.54	1824	1284	391	328	81.21

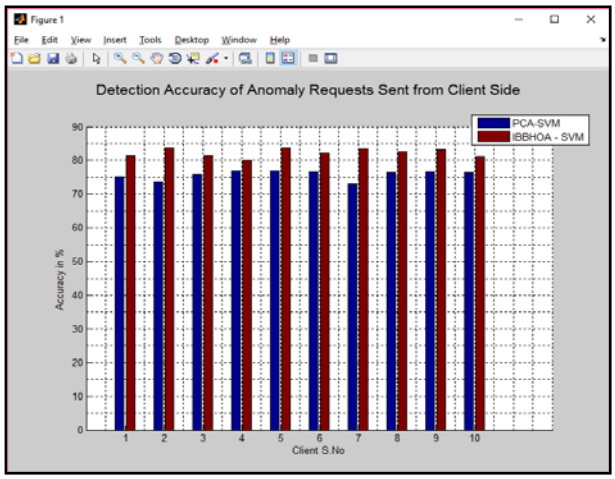

Fig.1. MATLAB Result for Detection Accuracy of Anomaly Requests Sent from Client Side

Table IV Detection Accuracy of Anomaly Requests Received – At Server Side

Domain S. No.	PCA - SVM Classifier					IBBHOA – SVM Classifier				
	TP	TN	FP	FN	Accuracy	TP	TN	FP	FN	Accuracy
1	52894	26052	12054	11749	76.83	54863	29592	9842	8452	82.20
2	34794	34191	11892	10959	75.12	36045	39452	8185	8154	82.21
3	41046	21449	10945	11342	73.71	44862	23854	8524	7542	81.05
4	21562	19093	7934	6147	74.27	23419	21034	5308	4975	81.21
5	22703	23503	7945	6887	75.70	25893	24503	6002	4640	82.56
6	26566	23543	7933	6912	77.15	29475	23912	6812	4754	82.19
7	35748	35428	8844	12723	76.75	38663	36582	7501	9997	81.13
8	31095	37846	12034	9689	76.04	33801	40371	8991	7501	81.81
9	37125	29642	9948	9783	77.19	39854	31154	6235	9255	82.09
10	35023	30543	10339	9148	77.09	38029	31167	7952	7905	81.36

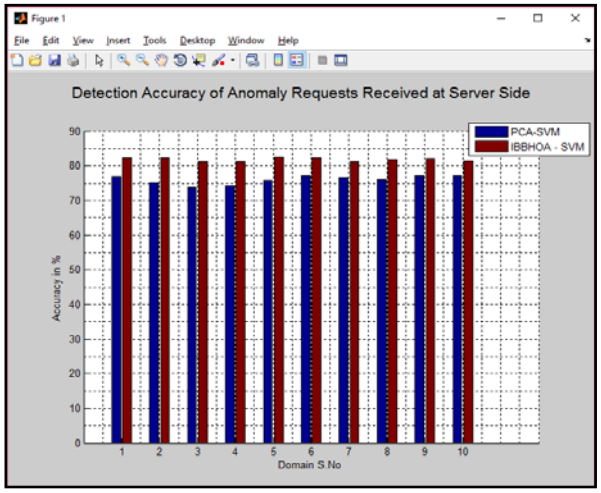

Fig.2. MATLAB Result for Detection Accuracy of Anomaly Requests Received – At Server Side

Table V Time Taken for Feature Selection and Classification – Client side

S.No.	IP Address	Queries / Day	Feature Selection		Classification	
			PCA-SVM	IBBHOA-SVM	PCA-SVM	IBBHOA-SVM
1	139.59.17.152	2,41,056	14.46	6.51	21.70	18.32
2	59.99.150.200	2,28,159	13.69	6.16	20.53	17.34
3	223.179.220.177	1,14,311	6.86	3.09	10.29	8.69
4	118.151.209.5	95,981	5.76	2.59	8.64	7.29
5	203.90.4.145	84,452	5.07	2.28	7.60	6.42
6	103.234.190.183	74,191	4.45	2.00	6.68	5.64
7	122.176.20.6	73,445	4.41	1.98	6.61	5.58
8	150.107.25.248	65,049	3.90	1.76	5.85	4.94
9	112.133.201.135	62,237	3.73	1.68	5.60	4.73
10	103.1.115.219	52,617	3.16	1.42	4.74	4.00

Table VI Time Taken for Feature Selection and Classification – Server side

S.No.	Domain Name	Queries / Day	Feature Selection		Classification	
			PCA-SVM	IBBHOA-SVM	PCA-SVM	IBBHOA-SVM
1	baidu.com	37,89,755	227.39	6.14	341.08	288.02
2	qq.com	26,61,189	159.67	4.31	239.51	202.25
3	akadns.net	23,87,316	143.24	3.87	214.86	181.44
4	apple.com	17,90,093	107.41	2.90	161.11	136.05
5	taobao.com	11,86,935	71.22	1.92	106.82	90.21
6	in-addr.arpa	10,89,644	65.38	1.77	98.07	82.81
7	google.com	9,81,895	58.91	1.59	88.37	74.62
8	weibo.com	7,98,301	47.90	1.29	71.85	60.67
9	sina.com.cn	7,85,648	47.14	1.27	70.71	59.71
10	360.cn	7,55,655	45.34	1.22	68.01	57.43

Table 1.presents top 10 wireless client IP addresses and theTable 2. shows top 10 destination domain names. Table 3.portrays detection accuracy of anomaly requests sent from client side. From the Table 3 it is evident that the proposed IBBHOA based SVM detect more anomaly behavior in wireless clients than that of PCA-SVM. It is noteworthy that from the Table 4 it is evident that the proposed IBBHOA based SVM detect more anomaly behavior in domain name servers than that of PCA-SVM. Table 5 encompasses the time taken for performing feature selection task and time taken for performing classification task. It is evident that the proposed mechanism outperforms the existing one.

5. Conclusion

This research work aims to detect anomalies in DNS query logs. The data are obtained from CAIDA data server [23]. For that reason, the proposed work introduces binding approach based on IBBHOA for feature selection in SVM classifier. Initially unremitting black hole optimization algorithm is portrayed. After that an improved binary black hole optimization algorithm is presented. Then the binding approach based on IBBHOA for feature selection in SVM Classifier is carried out. SVM classifier is chosen for performing the classification task for characterizing DNS lookup behaviors by means of log-mining. Feature selection is performed and then the SVM classifier is used to classify anomaly behaviors, time taken for feature selection and time taken for classification are the performance metrics chosen for comparison. From the obtained results, it is evident that the proposed mechanism outperforms than that of the existing one.

References

[1] A. Berger, W.N. Gansterer, Modeling DNS agility with DNSMap, in: Proceedings of IEEE INFOCOM Workshop on Traffic Monitoring and Analysis, Turin, Italy, 2013.
[2] A. Bouloutas, G. Hart, and M. Schwartz, “On the design of observers for failure detection of discrete event systems,” in IEEE Workshop on Network Management and Control, Tarrytown, NY, September 1989.
[3] D. Lee and M. Yannakakis, “Principles and methods of testing finite state machines-a survey,” Proceedings of the IEEE, vol. 84, no. 8, pp.1090–1123, 1996.
[4] R. E. Miller, “Passive testing of networks using a cfsm specification,” in Performance, Computing and Communications, 1998. IPCCC’98., IEEE International. IEEE, 1998, pp. 111–116.
[5] R. E. Miller and K. A. Arisha, “On fault location in networks by passive testing,” in Performance, Computing, and Communications Conference, 2000. IPCCC’00.Conference Proceeding of the IEEE International. IEEE, 2000, pp. 281–287.

[6] D. Whyte, E. Kranakis and P. C. Oorschot, "DNS-based Detection of Scanning Worms in an Enterprise Network," Network and Distributed System Security Symposium, 2005.
[7] D. Ludenar, H. Nagatomi, Y. Musashi, R. Matsuba, and K. Sugitani, "A DNS-based Countermeasure Technology far Bot Worm-infected PC terminals in the Campus Network," Journal for Academic Computing and Networking, 2006.
[8] N. Chatzis, "Mass MailingWonn Detection by Means of Situ-ation Aware DNS," International Symposium on Autonomous Decentralized Systems, 2007.
[9] P. Ren, J. Kristo, B. Gooch, Visualizing DNS traffic, in: Proceedings of the 3rd International Workshop on Visualization for Computer Security, ACM, New York, NY, USA, 2006, pp. 23–30.
[10] S. Hadi, S. Ali, Ali A. Ghorbani, A survey of visualization systems for network security, IEEE Trans. Vis. Comput.Graph. 18 (8) (2012) 1313–1329.
[11] A. Schonewille, D.J. van Helmond. The domain name service as an IDS. Research Project for the Master System-and Network Engineering at the University of Amsterdam, 2006.
[12] P. David, B. Paul, Context-aware clustering of DNS query traffic, in: Proceedings of the 8th ACM SIGCOMM Conference on Internet Measurement, ACM, New York, NY, USA, 2008, pp. 217–230.
[13] K. Bonnie, L. Simon, X. Wei, Analyzing Root DNS Traffic ⟨http://www.eecs.berkeley.edu/bbkirk/papers/cs262a-2004.pdf⟩, 2004.
[14] G.H. Shan, Y. Wang, M.J. Xie, H.P. Lv, X.B. Chi, Visual detection of anomalies in DNS query log data. In: 2014 IEEE Pacific Visualization Symposium (PacificVis), IEEE, New York, NY, USA, 2014, pp. 258–261.
[15] Conrad Albrecht-buehler, Benjamin Watson, David A Shamma, Visualizing live text streams using motion and temporal pooling. IEEE Comput. Graph. Appl. 25 (3) (2005) 52–59.
[16] UlrikBrandes, Daniel Fleischer, Jrgen Lerner, Highlighting conflict dynamics in event data, in: IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.IEEE, 2005. New York, NY, USA, pp. 103–110.
[17] Pieter Lexis, MatthijsMekking, Identifying Patterns in DNS Traffic, 2013.
[18] H. Yu, et al., A visualization analysis tool for DNS amplification attack, in: 2010 3rd International Conference on Biomedical Engineering and Informatics (BMEI), IEEE, New York, NY, USA, 2010, pp. 2834–2838.
[19] Kenton Born, David Gustafson, Ngviz: detecting dns tunnels through n-gram visualization and quantitative analysis, in: Proceedings of the Sixth Annual Workshop on Cyber Security and Information Intelligence Research, ACM, New York, NY, USA, 2010, p. 47.
[20] Mockapetris, P.: Domain Names Implementation and Specification. RFC 1035 (November 1987).
[21] Internet Engineering Task Force: Requirements for Internet Hosts Application and Support. RFC 1123 (October 1989).
[22] Vixie, P.: Extension Mechanisms for DNS (EDNS0). RFC 2671 (August 1999).
[23] CAIDA server: http://data.caida.org/datasets/dns/root-gtld-rtt/data/tokyo/2009/09/

AUTHOR PROFILE

S.S.Suganya working as an Assistant Professor, Department of Computer Science at Dr.SNSRajalakshmi College of Arts and Science, Coimbatore ,Tamil Nadu, India. She is having 10 Years of teaching experience. She attended various National and International Conferences and published 2 articles in international journal. Her area of specialization is Data mining and warehousing.