Eurasian Journal of Analytical Chemistry
ISSN: 1306-3057 OPEN ACCESS 2017 12 (4): 101-110

A Novel Group Testing (GT) - based Approach Deployed on Back-End Servers

G. Michael, R. Kavitha

Received: 06 December 2016 ▪ Revised: 09 January 2017 ▪ Accepted: 08 February 2017

Abstract: Application dos attack, which aims at disrupting application service rather than depleting the network resource, has emerged as a larger threat to network services, compared to the classic dos attack. Owing to its high similarity to legitimate traffic and much lower launching overhead than classic ddos attack, this new assault type cannot be efficiently detected or prevented by existing detection solutions. To identify application dos attack, we propose a novel group testing (GT)-based approach deployed on back-end servers, which not only offers a theoretical method to obtain short detection delay and low false positive/negative rate, but also provides an underlying framework against general network attacks.core specifically, we first extend classic GT model with size constraints for practice purposes, then redistribute the client service requests to multiple virtual servers embedded within each back-end server machine, according to specific testing matrices. Based on this framework, we propose a two-mode detection mechanism using some dynamic thresholds to efficiently identify the attackers. The focus of this work lies in the detection algorithms proposed and the corresponding theoretical complexity analysis.

Keywords: Back-End Servers, Novel Group Testing (GT), DENIAL-OF-SERVICE (DoS), Classic Methods.

INTRODUCTION

Prologue

DENIAL-OF-SERVICE (DoS) attack, which aims to make a service unavailable to legitimate clients, has become a severe threat to the Internet security. Traditional DoS attacks mainly abuse the network bandwidth around the Internet subsystems and degrade the quality of service by generating congestions at the network. Consequently, several network-based defense methods have tried to detect these attacks by controlling traffic volume or differentiating traffic patterns at the intermediate routers. However, with the boost in network bandwidth and application service types, recently, the target of DoS attacks has shifted from network to server resources and application procedures themselves, forming a new application DoS attack.

As stated in, by exploiting flaws in application design and implementation, application DoS attacks exhibit three advantages over traditional DoS attacks which help evade normal detections: malicious traffic is always indistinguishable from normal traffic, adopting automated script to avoid the need for a large amount of "zombie" machines or bandwidth to launch the attack, much harder to be traced due to multiple redirections at proxies. According to these characteristics, the malicious traffic can be classified into legitimate-like requests of two cases:

- At a high inter arrival rate.
- Consuming more service resources.

We call these two cases "high-rate" and "high-workload" attacks, respectively, in this paper.

Since these attacks usually do not cause congestion at the network level; thus, bypass the network-based monitoring system, detection, and mitigation at the end system of the victim servers have been proposed. Among them, the DDoS shield and CAPTCHA-based defense are the representatives of the two

G. Michael, Associate Professor, Department of Computer Science and Engineering, BIST, BIHER, Bharath Institute of Higher Education & Research, Selaiyur, Chennai. E-mail: michaelcse@gmail.com

R. Kavitha, Associate Professor, Department of Computer Science and Engineering, BIST, BIHER, Bharath Institute of Higher Education & Research, Selaiyur, Chennai.

major techniques of system-based approaches: session validation based on legitimate behavior profile and authentication using human-solvable puzzles. By enhancing the accuracy of the suspicion assignment for each client session, DDoS shield can provide efficient session schedulers for defending possible DDoS attacks. However, the overhead for per-session validation is not negligible, especially for services with dense traffic. CAPTCHA-based defenses introduce additional service delays for legitimate clients and are also restricted to human interaction services.

A kernel observation and brief summary of our method is: the identification of attackers can be much faster if we can find them out by testing the clients in group instead of one by one. Thus, the key problem is how to group clients and assign them to different server machines in a sophisticated way, so that if any server is found under attack, we can immediately identify and filter the attackers out of its client set. Apparently, this problem resembles the group testing (GT) theory which aims to discover defective items in a large population with the minimum number of tests where each test is applied to a subset of items, called pools, instead of testing them one by one. Therefore, we apply GT theory to this network security issue and propose specific algorithms and protocols to achieve high detection performance in terms of short detection latency and low false positive/negative rate. Since the detections are merely based on the status of service resources usage of the victim servers, no individually signature-based authentications or data classifications are required; thus, it may overcome the limitations of the current solutions.

GT was proposed during World War Two and has been applied to many areas since then, such as medical testing, computer networks, and molecular biology. The advantages of GT lie in its prompt testing efficiency and fault-tolerant decoding methods. To our best knowledge, the first attempts to apply GT to networking attack defense are proposed in parallel by Thai et al. (which is the preliminary work of this journal) and Khattabet al. The latter proposed a detection system based on "Reduced-Randomness Non adaptive Combinatorial Group Testing". However, since this method only counts the number of incoming requests rather then monitoring the server status, it is restricted to defending high-rate DoS attacks and cannot handle high-workload ones.

In a system viewpoint, our defense scheme is to embed multiple virtual servers within each physical back-end server and map these virtual servers to the testing pools in GT, then assign clients into these pools by distributing their service requests to different virtual servers. By periodically monitoring some indicators (e.g., average responding time) for resource usage in each server and comparing them with some dynamic thresholds, all the virtual servers can be judged as "safe" or "under attack." By means of the decoding algorithm of GT, all the attackers can be identified. Therefore, the biggest challenges of this method are threefold:

- How to construct a testing matrix to enable prompt and accurate detection.
- How to regulate the service requests to match the matrix in practical system.
- How to establish proper thresholds for server source usage indicator, to generate accurate test outcomes.

Similar to all the earlier applications of GT, this new application to network security requires modifications of the classical GT model and algorithms, so as to overcome the obstacle of applying the theoretical models to practical scenarios. Specifically, the classical GT theory assumes that each pool can have as many items as needed and the number of pools for testing is unrestricted. However, in order to provide real application services, virtual servers cannot have infinite quantity or capacity i.e., constraints on these two parameters are required to complete our testing model. Our main contributions in this paper are as follows:

- Propose a new size-constrained GT model for practical DoS detection scenarios.
- Provide an end-to-end underlying system for GT based schemes, without introducing complexity at the network core.
- Provide multiple dynamic thresholds for resource usage indicators, which help avoid error test from legitimate bursts and diagnose servers handling various amount of clients.
- Present three novel detection algorithms based on the proposed system, and show their high efficiencies in terms of detection delay and false positive/negative rate via theoretical analysis and simulations.

Besides application DoS attacks, our defense system is applicable to DoS attacks on other layers, e.g., protocol layer attack SYN flood where victim servers are exhausted by massive half-open connections. Although these attacks occur in different layers and of different styles, the victim machines will gradually run out of service resource and indicate anomaly. Since our mechanism only relies on the feedback of the victims, instead of monitoring the client behaviors or properties, it is promising to tackle these attack types.

Classical Group Testing Model

The classic GT model consists of t pools and n items (including at most d positive ones).As shown in below figure 1.2.1. This model can be represented by a t x n binary matrix M where rows represent the pools and columns represent the items. An entry M[i,j]=1 if and only if the ith pool contains the jth item; otherwise, M[i,j]=0. The t-dimensional binary column vector V denotes the test outcomes of these t pools, where 1-entry represents a positive outcome and 0-entry represents a negative one. Note that a positive outcome indicates that at least one positive item exists within this pool, whereas negative one means that all the items in the current pool are negative.

$$M = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix} \overset{\text{testing}}{\Longrightarrow} V = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Fig. 1: Binary testing matrix M and testing outcome vector V.

Classic Methods

Two traditional GT methods are adaptive and nonadaptive. Adaptive methods, a.k.a. sequential GT, use the results of previous tests to determine the pool for the next test and complete the test within several rounds. While nonadaptive GT methods employ d-disjunct matrix, run multiple tests in parallel, and finish the test within only one round. We investigate both these methods and propose three algorithms accordingly.

Decoding Algorithms

For sequential GT, at the end of each round, items in negative pools are identified as negative, while the ones in positive pools require to be further tested. Notice that one item is identified as positive only if it is the only item in a positive pool.

Nonadaptive GT takes d-disjunct matrices as the testing matrix M, where no column is contained in the Boolean summation of any other d columns. Du and Hwang proposed a simple decoding algorithm for this matrix type. A sketch of this algorithm can be shown using Fig. 1 as an example. Outcomes V [3] and V [4] are 0, so items in pool 3 and pool 4 are negative, i.e., items 3, 4, and 5 are negative. If this matrix M is a d-disjunct matrix, items other than those appearing in the negative pools are positive; therefore, items 1 and 2 are positive ones.

Apply to Attack Detection

A detection model based on GT can be assume that there are t virtual servers and n clients, among which d clients are attackers. Consider the matrix Mtxn in Fig. 1, the clients can be mapped into the columns and virtual servers into rows in M, where M [i, j]=1 if and only if the requests from client j are distributed to virtual server i.

With regard to the test outcome column V, we have V [i]=1 if and only if virtual server i has received malicious requests from at leastone attacker, but we cannot identify the attackers at once unless this virtual server is handling only one client. Otherwise, if V [i]=0, all the clients assigned to server i are legitimate. The d attackers can then be captured by decoding the outcome vector V and the matrix M.

Attacker Model

The maximum destruction caused by the attacks includes the depletion of the application service resource at the server side, the unavailability of service access to legitimate user, and possible fatal system errors which require rebooting the server for recovery. We assume that any malicious behaviors can be discovered by monitoring the service resource usage, based on dynamic value thresholds over the monitored objects.

Data manipulation and system intrusion are out of this scope. Similar to , we assume that application interface presented by the servers can be readily discovered and clients communicate with the servers using HTTP/1.1 sessions on TCP connections. We consider a case that each client provides a non spoofed ID (e.g., SYN-cookie) which is utilized to identify the client during our detection period. Despite that the application DoS attack is difficult to be traced, by identifying the IDs of attackers, the firewall can block the subsequent malicious requests.

As mentioned in Section 1, the attackers are assumed to launch application service requests either at high inter arrival rate or high workload, or even both. The term "request" refers to either main request or embedded request for HTTP page.

We further assume that the number of attackers d << n where n is the total client amount. This arises from the characteristics of this attack. Due to the benefits of virtual Servers we employ, this constraint can be relaxed, but we keep it for the theoretical analysis in the current work.

Victim/Detection Model

The victim model in our general framework consists of multiple back-end servers, which can be Web/application servers, database servers, and distributed file systems. We do not take classic multitier Web servers as the model, since our detection scheme is deployed directly on the victim tier and identifies the attacks targeting at the same victim tier; thus, multitier attacks should be separated into several classes to utilize this detection scheme. The victim model along with front-end proxies is shown in Fig.1.4. We assume that all the back-end servers provide multiple types of application services to clients using HTTP/1.1 protocol on TCP connections. Each back-end server is assumed to have the same amount of resource.

Moreover, the application services to clients are provided by K virtual private servers (K is an input parameter), which are embedded in the physical back-end server machine and operating in parallel. Each virtual server is assigned with equal amount of static service resources, e.g., CPU, storage, memory, and network bandwidth. The operation of any virtual server will not affect the other virtual servers in the same physical machine.

The reasons for utilizing virtual servers are twofold: first, each virtual server can reboot independently, thus is feasible for recovery from possible fatal destruction; second, the state transfer overhead for moving clients among different virtual servers is much smaller than the transfer among physical server machines.

As soon as the client requests arrive at the front-end proxy, they will be distributed to multiple back-end servers for load balancing, whether session sticked or not. Notice that our detection scheme is behind this front-end tier, so the load balancing mechanism is orthogonal to our setting. On being accepted by one physical server, one request will be simply validated based on the list of all identified attacker IDs (black list).

If it passes the authentication, it will be distributed to one virtual servers within this machine by means of virtual switch.

This distribution depends on the testing matrix generated by the detection algorithm. By periodically monitoring the average response time to service requests and comparing it with specific thresholds fetched from a legitimate profile, each virtual server is associated with a "negative" or "positive" outcome.

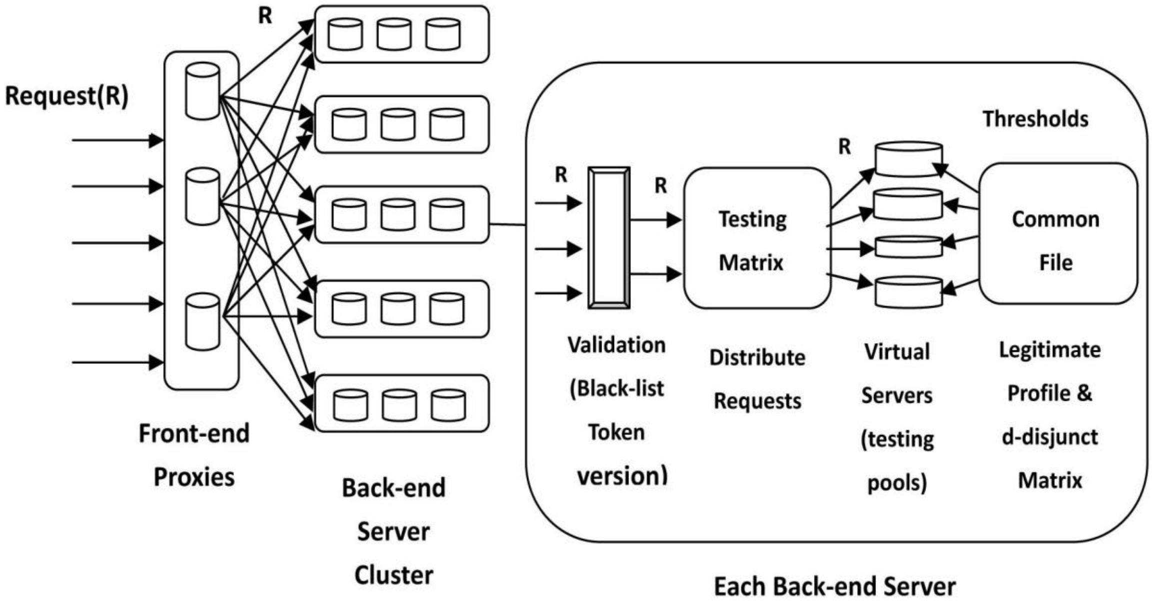

Fig.1: Victim/detection model

SYSTEM DESIGN

System Architecture

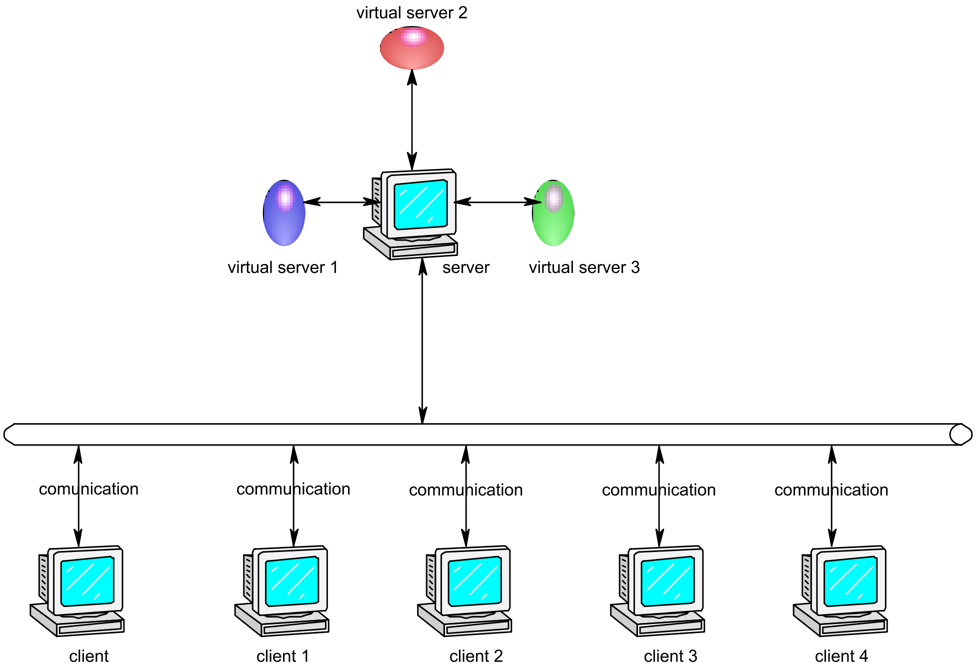

Fig. 2: Overview of System Architecture

Detailed Architecture Diagram

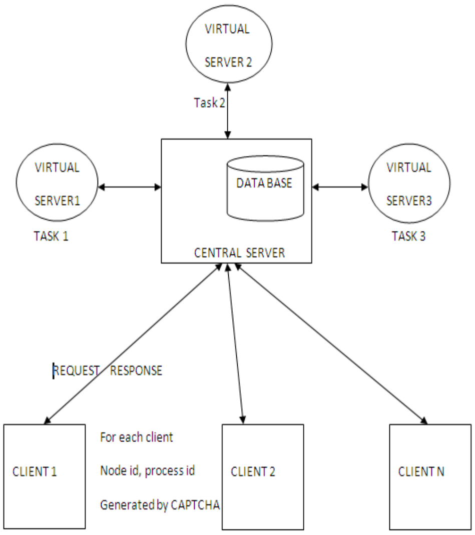

Fig. 3: Detailed Architecture Diagram

Contex Diagram

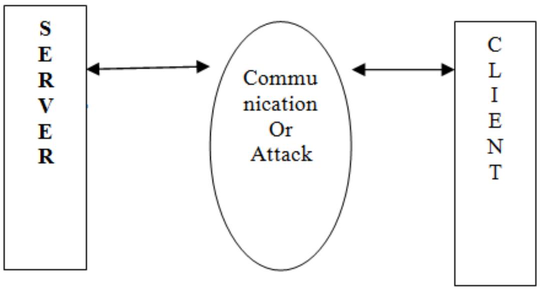

Fig. 4: Context Diagram

Data Flow Diagram

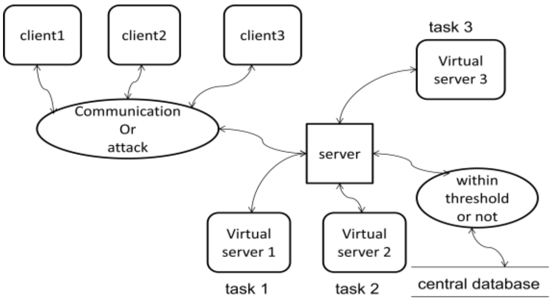

Fig. 5: DFD

Modules Description

1) Node Details Declaration
2) Server Creation
3) Server Monitoring
4) Captcha Generation

1. Node Details Declaration

In node details declaration, first node get register to network topology, by specifying the node IP address, Port Number and status. Node login to the network topology while sever will check the user authentication. Then only server system, allows the node in to the transmission .Node can send the packets to the destination or otherwise can send to server system. Node can add and relive is very easy in the network. Status also monitor by server system.

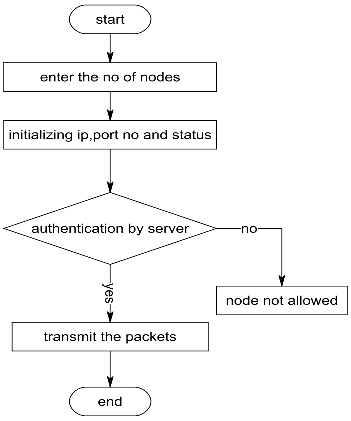

Fig. 6: Node Declaration

2. Server Creation

In server creation, the centralized server system will be designed for whole network. It has one centralized database and collect the details of each node. And store in to the centralized database. Server maintains these details, it very useful for node calculation and node details identification. Server can receive the request from all clients and the provide the corresponding response.

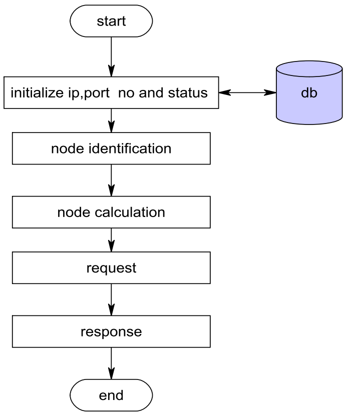

Fig. 7: Server Creation

3. Server Monitoring

In Server monitoring if we have any problem in network, server will be take the necessary action. The action is to not only discard the particular packet but also sever will collect the details about that particular node from database then the particular node is removed from the network. Server system can identify the node by using the captcha mechanism. Monitoring process also detect the attacker node in the whole network. Monitoring result also store in the server side.

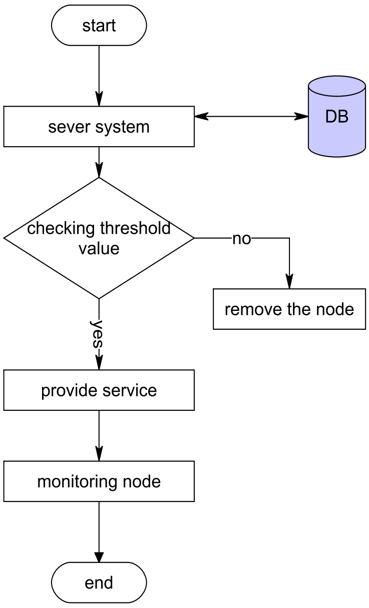

Fig. 8: Server Creation

4. Captcha Generation

In Captcha generation, each request is notified by using this unique captcha. This captcha is unique for all system. Captcha has two parts one is node-id and another one is process-id. Each node has the node-id as node name and port number combination. And each Process-id started from the process name and combine with process count. It used to identify the node and type of process from DOS attacking node

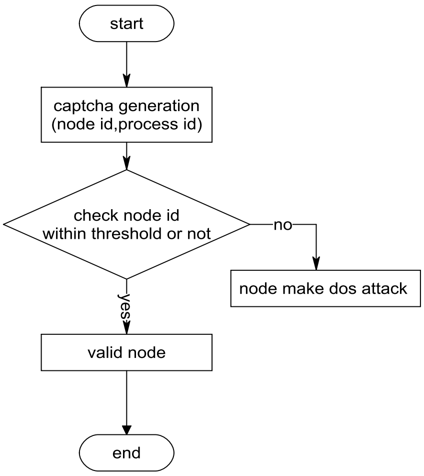
start
captcha generation (node id,process id)
check node id within threshold or not
no
node make dos attack
yes
valid node
end

Fig. 9: Captcha Generation

NODE DETAIL DECLARATION

Introduction

Denail Of Service (DoS) attack, which aims to make a service unavailable to legitimate clients, has become a severe threat to the Internet security.

Traditional DoS attacks mainly abuse the network bandwidth around the Internet subsystems and degrade the quality of service by generating congestions at the network. Consequently, several network-based defense methods havetried to detect these attacks by controlling traffic volume or differentiating traffic patterns at the intermediate routers.

However, with the boost in network bandwidth and application service types, recently, the target of DoS attacks has shifted from network to server resources and application procedures themselves, forming a new application DoS attack.

In node details declaration, first node get register to network topology, by specifying the node IP address, Port Number and status. Node login to the network topology while sever will check the user authentication. Then only server system allows the node into the transmission. Node can send the packet to the destination or otherwise can send to server system. Node can add and relieve is very easy in the network. Status also monitor by server system.

The centralized server system will be designed for whole network. It has one centralized database and collects the details of each node. And store in to the centralized database. Server maintains this details identification. Server can receive the request from all clients and provide the corresponding responses.

Architecture design

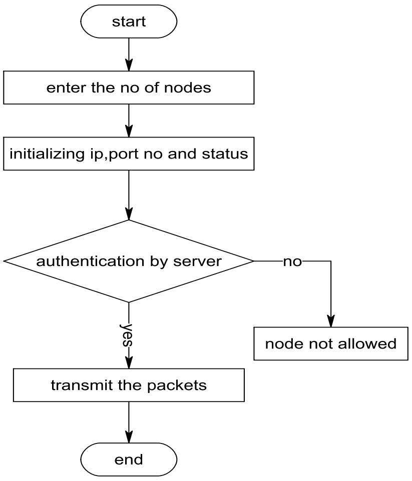

Fig. 10: Node Declaration

REFERENCES

[1] Das, J., Das, M. P., & Velusamy, P. (2013). Sesbania grandiflora leaf extract mediated green synthesis of antibacterial silver nanoparticles against selected human pathogens. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, *104*, 265-270.

[2] Umanath, K.P.S.S.K., Palanikumar, K., & Selvamani, S. T. (2013). Analysis of dry sliding wear behaviour of Al6061/SiC/Al2O3 hybrid metal matrix composites. *Composites Part B: Engineering*, *53*, 159-168.

[3] Udayakumar, R., Khanaa, V., Saravanan, T., & Saritha, G. (1786). Cross layer optimization for wireless network (WIMAX). *Middle-East Journal of Scientific Research*, *16*(12), 1786-1789.

[4] Kumaravel, A., & Rangarajan, K. (2013). Algorithm for automaton specification for exploring dynamic labyrinths. *Indian Journal of Science and Technology*, *6*(5S), 4554-4559.

[5] Pieger, S., Salman, A., & Bidra, A.S. (2014). Clinical outcomes of lithium disilicate single crowns and partial fixed dental prostheses: a systematic review. *The Journal of prosthetic dentistry*, *112*(1), 22-30.

[6] Vijayaraghavan, K., Nalini, S.K., Prakash, N.U., & Madhankumar, D. (2012). One step green synthesis of silver nano/microparticles using extracts of Trachyspermum ammi and Papaver somniferum. *Colloids and Surfaces B: Biointerfaces*, *94*, 114-117.

[7] Khanaa, V., Mohanta, K., & Satheesh, B. (2013). Comparative study of uwb communications over fiber using direct and external modulations. *Indian Journal of Science and Technology*, *6*(6), 4845-4847.

[8] Khanaa, V., Thooyamani, K. P., & Udayakumar, R. (1798). Cognitive radio based network for ISM band real time embedded system. *Middle-East Journal of Scientific Research*, *16*(12), 1798-1800.

[9] Vijayaraghavan, K., Nalini, S.K., Prakash, N.U., & Madhankumar, D. (2012). Biomimetic synthesis of silver nanoparticles by aqueous extract of Syzygium aromaticum. *Materials Letters*, *75*, 33-35

[10] Caroline, M.L., Sankar, R., Indirani, R.M., & Vasudevan, S. (2009). Growth, optical, thermal and dielectric studies of an amino acid organic nonlinear optical material: l-Alanine. *Materials Chemistry and Physics*, *114*(1), 490-494.

[11] Kumaravel, A., & Pradeepa, R. (2013). Efficient molecule reduction for drug design by intelligent search methods. *International Journal of Pharma and Bio Sciences*, *4*(2), B1023-B1029.

[12] Kaviyarasu, K., Manikandan, E., Kennedy, J., Jayachandran, M., Ladchumananandasiivam, R., De Gomes, U. U., & Maaza, M. (2016). Synthesis and characterization studies of NiO nanorods for enhancing solar cell efficiency using photon upconversion materials. *Ceramics International*, *42*(7), 8385-8394.

[13] Sengottuvel, P., Satishkumar, S., & Dinakaran, D. (2013). Optimization of multiple characteristics of EDM parameters based on desirability approach and fuzzy modeling. *Procedia Engineering*, *64*, 1069-1078.

[14] Anbuselvi S., Chellaram, C., Jonesh S., Jayanthi L., & Edward J.K.P. (2009). Bioactive potential of coral associated gastropod, Trochus tentorium of Gulf of Mannar, Southeastern India. *J. Med. Sci,* 9(5), 240-244.

[15] Kaviyarasu, K., Ayeshamariam, A., Manikandan, E., Kennedy, J., Ladchumananandasivam, R., Gomes, U. U., & Maaza, M. (2016). Solution processing of CuSe quantum dots: Photocatalytic activity under RhB for UV and visible-light solar irradiation. *Materials Science and Engineering: B*, *210*, 1-9.

[16] Kumaravel, A., & Udayakumar, R. (2013). Web portal visits patterns predicted by intuitionistic fuzzy approach. *Indian Journal of Science and Technology*, *6*(5S), 4549-4553.

[17] Srinivasan, V., & Saravanan, T. (2013). Reformation and market design of power sector. *Middle-East Journal of Scientific Research*, *16*(12), 1763-1767.

[18] Kaviyarasu, K., Manikandan, E., Kennedy, J., & Maaza, M. (2015). A comparative study on the morphological features of highly ordered MgO: AgO nanocube arrays prepared via a hydrothermal method. *RSC Advances*, *5*(100), 82421-82428.

[19] Kumaravel, A., & Udhayakumarapandian, D. (2013). Consruction of meta classifiers for apple scab infections. *International Journal of Pharma and Bio Sciences*, *4*(4), B1207-B1213.

[20] Sankari, S. L., Masthan, K. M. K., Babu, N. A., Bhattacharjee, T., & Elumalai, M. (2012). Apoptosis in cancer-an update. *Asian Pacific journal of cancer prevention*, *13*(10), 4873-4878

[21] Harish, B. N., & Menezes, G. A. (2011). Antimicrobial resistance in typhoidal salmonellae. *Indian journal of medical microbiology*, *29*(3), 223-229.

[22] Manikandan, A., Manikandan, E., Meenatchi, B., Vadivel, S., Jaganathan, S. K., Ladchumananandasivam, R., & Aanand, J. S. (2017). Rare earth element (REE) lanthanum doped zinc oxide (La: ZnO) nanomaterials: synthesis structural optical and antibacterial studies. *Journal of Alloys and Compounds*, *723*, 1155-1161.

[23] Caroline, M. L., & Vasudevan, S. (2008). Growth and characterization of an organic nonlinear optical material: L-alanine alaninium nitrate. *Materials Letters*, *62*(15), 2245-2248.

[24] Saravanan T., Srinivasan V., Udayakumar R. (2013). A approach for visualization of atherosclerosis in coronary artery, Middle - East Journal of Scientific Research, 18(12), 1713-1717.

[25] Poongothai, S., Ilavarasan, R., & Karrunakaran, C.M. (2010). Simultaneous and accurate determination of vitamins B1, B6, B12 and alpha-lipoic acid in multivitamin capsule by reverse-phase high performance liquid chromatographic method. *International Journal of Pharmacy and Pharmaceutical Sciences*, *2*(4), 133-139.

[26] Udayakumar, R., Khanaa, V., & Saravanan, T. (2013). Synthesis and structural characterization of thin films of SnO 2 prepared by spray pyrolysis technique. *Indian Journal of Science and Technology*, *6*(6), 4754-4757

[27] Anbazhagan, R., Satheesh, B., & Gopalakrishnan, K. (2013). Mathematical modeling and simulation of modern cars in the role of stability analysis. *Indian Journal of Science and Technology*, *6*(5S), 4633-4641.

[28] Caroline, M.L., & Vasudevan, S. (2009). Growth and characterization of bis thiourea cadmium iodide: A semiorganic single crystal. *Materials Chemistry and Physics*, *113*(2-3), 670-674.

[29] Sharmila, S., Jeyanthi Rebecca, L., & Das, M. P. (2012). Production of Biodiesel from Chaetomorpha antennina and Gracilaria corticata. *Journal of Chemical and Pharmaceutical Research*, *4*(11), 4870-4874.

[30] Meharban, M.S., & Dr. Priya, S. (2016). A Review on Image Retrieval Techniques. *Bonfring International Journal of Advances in Image Processing, 6*(2), 07-10.

[31] Ravindaranaath, R.J., Karthik, K., Vishnupriyan, R., Suryakumar, S., & Thamaraiselvi, G. (2017). Automated Trolley System for Airport. *International Journal of Communication and Computer Technologies, 5*(1), 32-35.

[32] Balamurugan, R., & Nagarajan, N.R. (2017). Automatic Robotic Arm Using Hand Gesture. *International Journal of Communication and Computer Technologies, 5*(2), 43-45.

[33] Mahendran, S. (2017). Fault Detection in Power Transmission Line. *International Journal of Communication and Computer Technologies, 5*(2), 46-47.

[34] Sebastian, J., & Anthony, D. (2018). Filtering and Summarization Architecture for News Pages. *Journal of Computational Information Systems, 12*(3), 1-7.

[35] Yathisha, L., Pavithra, A.C., & Shasidhar Gokhale, S. (2014). Novel Optimal LQR Switching Control Method for the Speed Control of DC Motor, *International Journal of Advances in Engineering and Emerging Technology, 5*(6), 248-257.

[36] Rajagopala Krishnan, N. (2014). Asynchronous FPGA Cellâ€Ÿs Design with Autonomous Power Gating and LEDR Encoding. *Excel International Journal of Technology, Engineering and Management, 1*(3), 84-90.

[37] Slimani, T. (2014). RST Approach for Efficient CARs Mining. *Bonfring International Journal of Data Mining, 4*(4), 34-40.

[38] Moradi, H., Namdaran, T., Gooran, P.R., Pouradad, E., & Aivazie, M.R. (2015). Design and Analysis of Equivalent Circuit Model Laser VSCEL Parameters Using Photonics. *International Academic Journal of Innovative Research, 2*(8), 20-37.

[39] Ramahrishnan, S., Geetha, B.E.R., & P. Vasuki, (2014). Image Encryption Using Chaotic Maps in Hybrid Domain. *International Journal of Communication and Computer Technologies, 2*(2), 74-78.

[40] Thooyamani, K.P., Khanaa, V., & Udayakumar, R. (2013). An integrated agent system for e-mail coordination using jade. *Indian Journal of Science and Technology*, *6*(6), 4758-4761.

[41] Caroline, M. L., Kandasamy, A., Mohan, R., & Vasudevan, S. (2009). Growth and characterization of dichlorobis l-proline Zn (II): A semiorganic nonlinear optical single crystal. *Journal of Crystal Growth*, *311*(4), 1161-1165.

[42] Caroline, M.L., & Vasudevan, S. (2009). Growth and characterization of L-phenylalanine nitric acid, a new organic nonlinear optical material. *Materials Letters*, *63*(1), 41-44.

[43] Kaviyarasu, K., Xolile Fuku, Genene T. Mola, E. Manikandan, J. Kennedy, and M. Maaza. Photoluminescence of well-aligned ZnO doped CeO2 nanoplatelets by a solvothermal route. *Materials Letters*, *183*(2016), 351-354.

[44] Saravanan, T., & Saritha, G. (2013). Buck converter with a variable number of predictive current distributing method. *Indian Journal of Science and Technology*, *6*(5S), 4583-4588.

[45] Parthasarathy, R., Ilavarasan, R., & Karrunakaran, C. M. (2009). Antidiabetic activity of Thespesia Populnea bark and leaf extract against streptozotocin induced diabetic rats. *International Journal of PharmTech Research*, *1*(4), 1069-1072.

[46] Hanirex, D. K., & Kaliyamurthie, K. P. (2013). Multi-classification approach for detecting thyroid attacks. *International Journal of Pharma and Bio Sciences,* 4(3), B1246-B1251

[47] Kandasamy, A., Mohan, R., Lydia Caroline, M., & Vasudevan, S. (2008). Nucleation kinetics, growth, solubility and dielectric studies of L-proline cadmium chloride monohydrate semi organic nonlinear optical single crystal. *Crystal Research and Technology: Journal of Experimental and Industrial Crystallography*, *43*(2), 186-192.

[48] Srinivasan, V., Saravanan, T., Udayakumar, R., & Saritha, G. (2013). Specific absorption rate in the cell phone user's head. *Middle-East Journal of Scientific Research*, *16*(12), 1748-50.

[49] Udayakumar R., Khanaa V., & Saravanan T. (2013). Chromatic dispersion compensation in optical fiber communication system and its simulation, Indian Journal of Science and Technology, 6(6), 4762-4766.

[50] Vijayaragavan, S.P., Karthik, B., Kiran, T.V.U., & Sundar Raj, M. (1990). Robotic surveillance for patient care in hospitals. *Middle-East Journal of Scientific Research*, *16*(12), 1820-1824.