International Journal of Electrical and Computer Engineering (IJECE)
Vol.10, No.1, February2020, pp. 997~1005
ISSN: 2088-8708, DOI: 10.11591/ijece.v10i1.pp997-1005 

# Detecting malicious URLs using binary classification through ada boost algorithm

**Firoz Khan[1], Jinesh Ahamed[2], Seifedine Kadry[3], Lakshmana Kumar Ramasamy[4]**
[1,2]Computer Information Science Department, Higher Colleges of Technology, UAE
[3]Department of Mathematics and Computer Science, Faculty of Science, Beirut Arab University, Lebanon
[4]Department of Computer Applications, Hindusthan College of Engineering and Technology, India

**Article Info**

***Article history:***

Received Apr 14, 2019
Revised Jun 29, 2019
Accepted Sep 27, 2019

***Keywords:***

AdaBoost algorithm
Binary classification problem
Blacklists
Machine learning
Malicious uniform resource locator

**ABSTRACT**

Malicious Uniform Resource Locator (URL) is a frequent and severe menace to cybersecurity. Malicious URLs are used to extract unsolicited information and trick inexperienced end users as a sufferer of scams and create losses of billions of money each year. It is crucial to identify and appropriately respond to such URLs. Usually, this discovery is made by the practice and use of blacklists in the cyber world. However, blacklists cannot be exhaustive, and cannot recognize zero-day malicious URLs. So to increase the observation of malicious URL indicators, machine learning procedures should be incorporated. In this study, we have developed a complete prototype of Malicious URL Detection using machine learning methods. In particular, we have attempted an exact formulation of Malicious URL exposure from a machine learning perspective and proposed an approach using the AdaBoost algorithm - the proposed approach has brought forward more accuracy than other existing algorithms.



***Corresponding Author:***

Seifedine Kadry,
Department of Mathematics and Computer Science,
Beirut Arab University,
Beirut, Lebanon.
Email: s.kadry@bau.edu.lb

## 1. INTRODUCTION

The arrival of modern intelligence technologies brings an enormous influence in the increase and advancement of markets over several applications. In the current age, it is nearly mandatory for an organization to have an online presence to have a successful and prosperous enterprise. As a consequence, the incorporation of the World Wide Web and Internet into the operations of the organization becomes essential. Regrettably, the technological progressions come with security issues that are used to trick and scam end users. Such attacks consist of illegal websites that market forged goods, commercial fraud conducted by cheating users into sharing delicate data with the ultimate aim to steal money or identification, or also planting a bad piece of code, malware, in the user's machine. As there are a diverse variety of attacks possible, and the various contexts in which such attacks can arise, it is difficult to invent robust operations to identify cyber-security crimes. The boundaries of conventional security administration technologies are growing more profound and addressing this exponential increase of current security menaces with the help of accelerated advances in modern IT technologies. However, there is a notable deficit of security specialists who can address this significant concern. The most significant of these attacking methods are identified by increasing compromised URLs [1].

URL is the acronym of Uniform Resource Locator, which signifies the Universal location of records and other sources on the World Wide Web. It has two principal elements: (i) protocol identifier, and it symbolizes what rules to apply, (ii) resource name, it defines the IP address or the domain name where

the source is situated. A colon and two forward slashes separate the protocol identifier and the resource name. A sample is displayed in Figure 1.

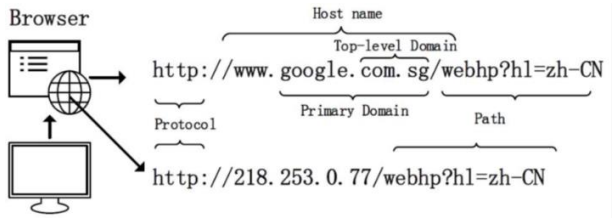


Figure 1. Uniform resource locator

Research has found that nearly one-third of all sites are probably malicious [2], confirming a significant use of malicious URLs to perform cyber-crimes. A URL of a malicious web site incorporates some type of free information to attract end users using spam, phishing, or drive-by-exploits to originate attacks. Innocent end users navigate the web sites and then gets victimized using several kinds of scams and malware. Modern archetypes of crimes practicing malicious URLs incorporate Drive-by Download, Phishing and Social Engineering, and Spam [3]. Drive-by-download [4] is defined as the accidental download of malware by sensing a URL. These attacks are typically conducted by employing vulnerabilities as add-ons or injecting a malicious piece of software written in JavaScript. Phishing and Social Engineering interventions [5] fool the end users by making them exchange personal or sensitive data by representing to be real web pages. Spam is the preferred method of volunteered communications used by criminals to promote these attacks or conduct phishing. These kinds of crimes happen in massive amounts and have produced billions of monetary value of loss each year. Efficient practices to identify such malicious URLs on time can effectively support to identify a substantial amount of and a kind of cyber-security warnings. In parallel, researchers and practitioners have acted on devising possible resolutions for Malicious URL Detection.

The rest of this article is organized as follows. Section 2 describes the work carried out by a few researchers. Section 3 outlines our methodology for solving the malicious classification. The experimental outcomes are put forth in Section 4. Concluding remarks is included in Section 5.

## 2. RELATED WORK

The usual standard way to identify malicious URLs by various antivirus software is the blacklist approach. All are principally a repository of URLs that have been verified to be malicious in history. This collection is verified across time and used to track if the URL is dangerous. Such a system is fast by behavior because of its simple query cost and therefore is very simple to execute. Moreover, such a procedure would produce minimal false-positive rates (Even though it has been found that usually blacklisting can endure non-trivial false-positive rates [6, 7]). However, it is nearly improbable to keep an exhaustive record of malicious URLs, particularly for distinct URLs which are created regularly. Criminals come up with inventive ways to avoid blacklists and trick end users by altering the URL to look authentic. Garera et al. [8] have classified four kinds of obfuscation: Obfuscating the Owner by an IP, Obfuscating the Host with a different domain, Obfuscating the host with great hostnames, and misspelling. Each of these methods tries to conceal the malicious purposes of the website by hiding the malicious URL.

In the current new trend and the expanding demand of URL shortening services, a unique and comprehensive obfuscation method is being observed [9, 10]. If the URLs seem reliable, and end user trusts the URL, an attack can be started. Usually, the criminals will further attempt to confuse the code to limit signature-based mechanisms from exposing them. Criminals do several additional routines to avoid blacklists through which proxy servers are automatically formed to host the web-page.Moreover, attackers usually begin further attacks, which changes the attack-signature, making it untraceable by devices that direct on particular signatures. Blacklisting systems have rigid boundaries, and it seems relevant to avoid them, mostly because blacklists are worthless for gaining forecasts on distinct URLs

The researchers Fossi et al., have developed an extensive collection of global threats that covers corporate data gaps, attacks on browsers and websites, spear phishing attempts, ransomware and different kinds of fraudulent cyber actions [11]. The research also reveals rare cyber skills practiced by the scammers. One efficient procedure is charging the users to click on a malicious Uniform Resource Locator (URL), which then makes the system arbitrated.

The web security group has begun blacklisting services to find malicious websites. These blacklists are formed by using multiple collections of systems including manual reporting, honeypots, and web crawlers united with locality investigation [12-13]. While URL blacklisting has remained satisfactory to any space, it is reasonably straightforward for an intruder to fool the system by lightly transforming one or more elements of the URL sequence. Furthermore, several malicious sites are not blacklisted because they are too new or were wrongly assessed. Studies conducted historically can be used to confirm this query from a Machine Learning viewpoint. That is, people collect a record of URLs that have remained categorized as either malicious or favorable and describe specific URL through a set of characteristics. Classification algorithms are then required to determine the line between the decision sets. The authors classified the undiscovered malicious Web sites by employing the characteristics of the network address [14-15]. The motive [16-17] of this research is to analyze malicious websites of good ones from their URL features. Through feature selection relating to Pareto GA, they delivered more precision and F-score with the most limited amount of features.

In [18], the authors suggested a solution to overcome the problem of embedding malware programs in the URLs by Machine Learning. The authors [19], made the conversation on the exploratory attack, which deceives the judgment of the classifier on the malicious units. The authors proposed a model perceived as the attack model, which is practised to attack the detection system utilizing Support Vector Machine and Fisher Discriminant Classifier. In [20], the work was performed by the authors to hold the spam action and further to remove the spam content and malicious URLs in Email. They have practised data mining approach, which increases the efficiency of the system and identifies more volume of spam and malicious URLs. DOS attack is an effort conceived by the intruder to refuse service to the user. It is an initiative that floods the victim system with traffic transmitting malicious information which may hit the system. The authors applied supervised learning algorithms Support Vector Machine and C4.5 on NSL_KDD Dataset for beneficial classification of DOS Attack [21]. Thus, it is necessary to develop a method of classifying malicious accesses automatically from various collected data, including both malicious and benign accesses. In this study, they have concentrated on the discovery of crawlers, whose accesses has been growing swiftly [22].

Phishing has progressed remarkably across the last few years, and it has shifted a critical warning to global security and economy. The authors aim to prove phishing discovery using fuzzy logic and interpreting outcomes using various defuzzification techniques [23]. Biao et al. [24] proposed a hierarchical clustering algorithm based on tree edit distance to recognize and categorize hostile JavaScript. A social networking site helps millions of users to interact online, and a substantial volume of data has been uploaded everyday. Hence in each second, a huge quantity of data has been caused throughout the world. This requires the adoption of the new methodology to provide security of online data. Social network users are not aware of the various security threats, and the associated risks exist in these networks. The authors present a methodology which supports the online users to be protected from various fraudulent and malicious actions on the network. This paper presents an assessment of classification different social network and different attacks present on those social networks and methodology has been proposed which help the online users to be safe from numerous fraudulent and malicious activities on the web [25].

## 3. METHODOLOGY

Boosting is a common approach for acquiring classifiers by converting weak learner to strong learner. The idea of the boosting technique is to get a weak classifier and practice it to develop an extremely beneficial classifier, through increasing the prediction of the weak classification algorithm. This prediction is prepared by equalizing the yields of many weak classifiers. One of the most popular boosting algorithms is AdaBoost, known as Adaptive Boosting that focusses on classification problems that build an influential classifier from many weak classifiers. It is arranged by developing a prototype from the training data, then building a second model that strives to fix the flaws from the first model. Models are appended until the training set is deduced, or the most number of models are combined.

AdaBoost is extremely practiced to increase the production of decision trees on binary classification problems. The reason for choosing the AdaBoost algorithm is that it can be practiced to increase the performance of any machine learning algorithm. It is always great when done among weak learners. The conventional algorithm applied with AdaBoost is decision trees with one level. The trees are small and hold one decision for a class, and hence are referred to as decision stumps. Weak models are combined sequentially, derived utilizing the weighted training data. The method continues until a pre-set quantity of weak learners has been produced, or no additional development can be made on the training dataset.

In terms of machine learning, in a decision tree, every node represents a search on an attribute. Every branch describes the result of the test, and the leaf nodes describe the class label received behind all judgments obtained within that branch. The ways from the root to leaf provides a classification rule. This system aims to represent the information while reducing the complexity of the design. Figure 2 shows the comparison between the unsupervised and supervised machine learning. The malicious and benign classifiers are further highlighted in the figure.

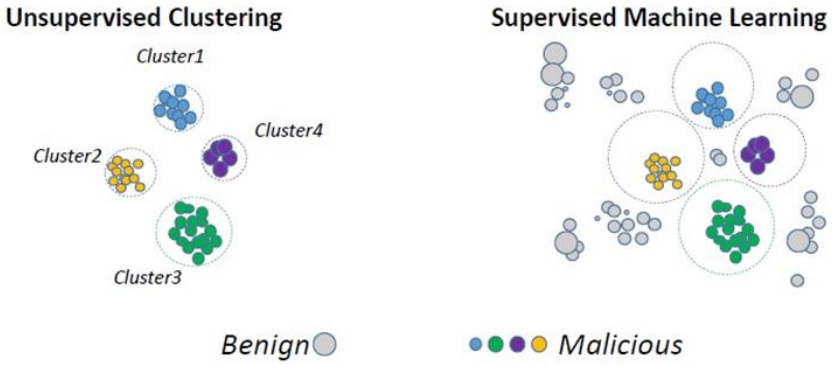


Figure 2. Machine learning comparison

The pseudocode of the AdaBoost algorithm is as follows: A weak classifier is taken through the training data using the samples. The AdaBoost algorithm supports only binary classification problems, so after each boosting iteration, the component classifier was discovered whose weighted error would be better than the previous one.

**Algorithm-1 AdaBoost algorithm**

```
function ADABOOSTING(samples, A, H) returns a weighted-majority hypothesis
  inputs: samples, set of N labeled samples (x1, y1),…,(xN,yN)
          A, an algorithm
          H, the hypotheses in the ensemble
  other variables: a, a vector of N example weights, initially 1/N
                   b, a vector of H hypotheses
                   z, a vector of H hypothesis weights
  for i = 1 to H do
      b[i] ← L(samples, a)
      error ← 0
      for j = 1 to N do
          if b[i](xj) ≠ yj then error ← error + a[j]
      for j = 1 to N do
          if b[i](xj) = yj then a[j] ← a[j] • error/(1 − error)
      a ← NORMALIZE(a)
      c[i] ← log(1 − error)/error
  return MAJORITY(b, c)
```

## 4. RESULTS AND DISCUSSIONS

For the experimental purpose, we have developed a web-based application with java platform, and the computers for this experiment have the same configuration: Intel Core-i7 2.40 GHz (4 CPU's), 8 GB RAM, Microsoft Windows 10 professional 64 bit, and JDK 1.7. The evidence for this research, presented by Ma et al. [13], contains around 121 sets of URLs, and each is classified as either malicious or benign. The whole dataset consists of over 2.3 million URLs, each producing over 3.2 million characteristics. The amazing majority of these peculiarities can be found as binary properties.The Malware classification chart was shown in Table 1.

Table 1. Malware classification chart

| Malware Type | Count |
|---|---|
| directs to Trojan | 1 |
| Torjan | 24 |
| RFI | 1 |
| compromised site/Redirects to Mebroot | 1 |
| Backdoor.Win32.KeyStart.m | 1 |
| exploits/mebroot | 13 |
| Mebroot calls home | 14 |
| Exlpoits | 11 |
| Redirects to exploits | 6 |
| Redirects to Mebroot | 3 |
| zeus v1 trojan | 23 |
| trojan downloader | 1 |
| zeus v1 (non-RC4) trojan | 19 |
| Asprox/Danmec | 13 |
| exploits/Trojans | 1 |
| trojan Waledac | 1 |
| redirects to Luckysploit | 1 |
| Fake Antivirus | 4 |
| IRCBot | 1 |
| redirects to mebroot and other exploits | 1 |
| Malware calls home | 2 |
| Luckysploit | 8 |
| exploits/Trojan | 3 |
| trojan vundo | 1 |
| zeus v1 (non-RC4) config file | 15 |
| zeus v1 config file | 10 |
| trojan clicker | 1 |
| zeus Trojan | 1 |

A prototype has been developed in Java platform using the AdaBoost algorithm [14, 15]. The screenshot below is attached as a reference. In Figure 3, a provision has been provided for the end users who use this platform, and they can use this screen to add the details of the malicious URL details if they encounter any. This provision was incorporated to give awareness to other end users.


Figure 3. Provision for adding the malicious site description

In Figure 4, a sample of Malicious URL was tested, and the classification was done. In the above Figure, we have tested a URL addressed http://free-best-movies.com/downloadvideo17637/index.html, and it was classified as Trojan by the proposed system. Additionally, we have tested the tool with many sites, and a few results have been given as an endpoint here. As we have planned to take this research ahead, we have not discussed our comprehensive details.


Figure 4. Verifying a phish

Figure 5 highlights the number of malware occurrences and our proposed systems classifies the malware into different categories like Directs to Trojan, Trojan, RFI, compromised site/Redirects to Mebroot, Backdoor.Win32.KeyStart.m, exploits/mebroot, Mebroot calls home, Exploits, redirects to exploits, Redirects to Mebroot, zeus v1, Trojan, trojan downloader, zeus v1 (non-RC4) trojan, Asprox/Danmec, exploits/Trojans, trojan Waledac, redirects to Luckysploit, Fake Antivirus, IRCBot , redirects to mebroot and other exploits , Malware calls home, Luckysploit, exploits/Trojan, trojan Vundo , zeus v1 (non-RC4) config file, zeus v1 config file, trojan clicker, zeus Trojan. A summary table of calssified malwares can be seen in Figure 6. The recommended method classifies if any new malware was identified. The model was trained and tested in such a way, and the testing file is mentioned below as in Figure 7.

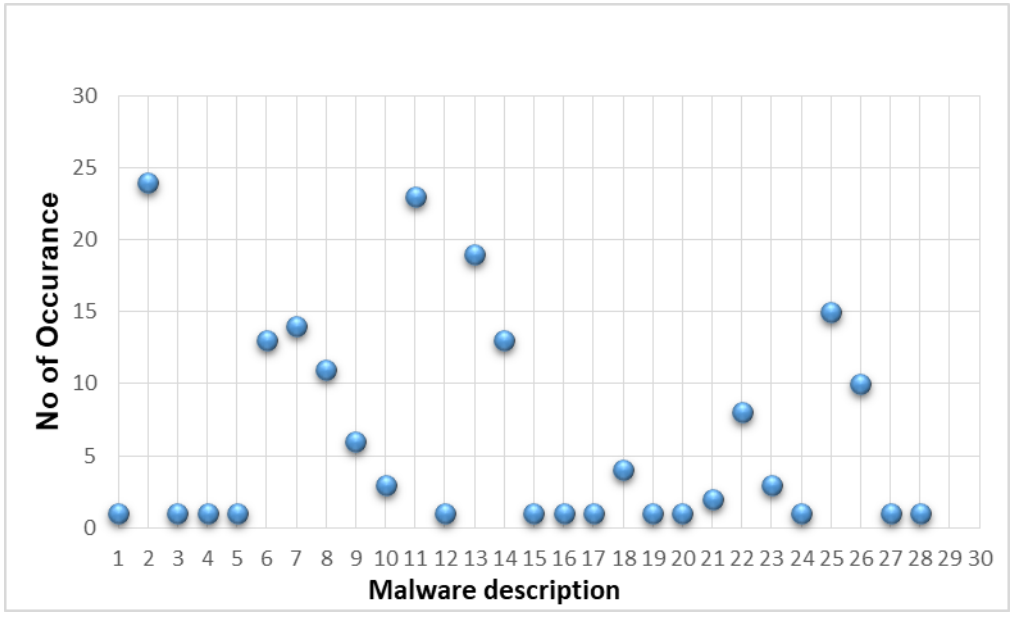


Figure 5. Occurrence of different malware

| Domain | IP | Reverse Lookup |
|---|---|---|
| free-best-movies.com/downloadvideo17637/index.html | 82.213.74.8 | vip-girls.biz |
| rightmoveit.co.uk/Photo/rg1.exe | 212.1.211.104 | srv211-104.hosting24.com. |
| webfo.biz/fbid1.txt | 69.89.27.211 | box211.bluehost.com |
| www.xtipp.hu/index.php_ | 195.70.48.68 | s5.mediacenter.hu |
| epery.com/wssl713fro.exe | 68.180.151.74 | p2p.geo.vip.sp1.yahoo.com |
| har5launo.com/cgi-bin/index.cgi?dx | 74.213.167.191 | 74-213-167-191.ultrahosting.com |
| hgcxianj.com | 216.55.163.216 | 216-55-163-216.dedicated.abac.net |
| hbjhejsc.com | 74.213.167.190 | 74-213-167-190.ultrahosting.com |
| epr.info | 85.17.136.137 | hosted-by.leaseweb.com |
| seocom.mobi/rotate/c.php?eb0h | 74.200.72.198 | unknown198.72.200.74.defenderhosting.com |
| www.368500.cn/vm/to.htm | 67.202.36.227 | ec2-67-202-36-227.compute-1.amazonaws.com |
| www.ouwou.cn/windows.exe | 216.245.209.196 | solar.lsn.servebyte.com |
| hz3launo.com/cgi-bin/index.cgi?dx | 74.213.167.191 | 74-213-167-191.ultrahosting.com |
| 222.2007wyt.net/sina.css | 219.153.71.185 | 185.71.153.219.broad.cq.cq.dynamic.163data.com.cn |
| bidwm.info/temp/load.php?id=318 | 77.221.137.138 | 77.221.137.138.addr.datapoint.ru |
| d.ko546.com/new/a1.css | 174.37.172.68 | 174.37.172.68-static.reverse.softlayer.com. |
| dhvhevg.com | 74.213.167.190 | 74-213-167-190.ultrahosting.com |
| hjfxwanj.com | 216.55.163.216 | 216-55-163-216.dedicated.abac.net |
| live-counter.net/load.php?id=6324871 | 79.113.45.174 | 79-113-45-174.rdsnet.ro |
| beta.jino-net.ru/agentvkontakte.exe | 217.107.217.29 | 29.0/27.217.107.217.in-addr.arpa. |
| www.highwrite.com/chinese/index.html | 174.133.178.194 | amg.mochahost.com. |
| www.patria-nova.de/user | 80.86.83.132 | pkreptiles.ch. |
| www.spcounter.info/count/load.php | 74.86.187.24 | 74.86.187.24-static.reverse.softlayer.com |
| www.tradingway.net/load.php?id=20603&spl=4 | 70.84.195.170 | aa.c3.5446.static.theplanet.com |
| zilya-sosal.com/fista/load.php?id=14737&spl=4 | 84.191.91.230 | 84-191-91-230.hostnoc.net |
| emralaunо.com/cgi-bin/index.cgi?dx | 74.213.167.191 | 74-213-167-191.ultrahosting.com |
| nohtingherez.cn/adv111.exe | 217.20.112.96 | subdivi.de |
| pharmacy-earth.com/file.php?q=4&w=PDF_IE | 78.159.102.97 | spryciarze.pl |
| copy-past.cn | 66.197.235.213 | chr1.dediboxes.co.uk |
| lovekills.ru/love/load.php?id=118&spl=4 | 84.16.228.146 | serverside.ru |
| 999999.ru/999999/dr.exe | 70.84.195.170 | aa.c3.5446.static.theplanet.com |

Figure 6. Classified malwares repository

```
@relation Test
@attribute Domain {https://www.google.co.in/ }
@attribute IP {172.217.163.163}
@attribute Des {directs_to_trojan, Trojan, RFI,
compromised_site/Redirects_to_Mebroot, Backdoor.Win32.KeyStart.m,
exploits/mebroot, Mebroot_calls_home, Exploits,
redirects_to_exploits, Redirects_to_Mebroot, zeus_v1_trojan,
trojan_downloader, zeus_v1_(non-RC4)_trojan, Asprox/Danmec,
exploits/trojans, trojan_Waledac, redirects_to_Luckysploit,
Fake_Antivirus, IRCBot, redirects_to_mebroot_and_other_exploits,
Malware_calls_home, Luckysploit, exploits/trojan, trojan_vundo,
zeus_v1_(non-RC4)_config_file, zeus_v1_config_file,
trojan_clicker, zeus_trojan, Exploit}
@data
https://www.google.co.in/,172.217.163.163,?
```

Figure 7. Testing data

## 5. CONCLUSION

Malicious URL discovery is a significant part of numerous cybersecurity applications, and it is evident that machine learning strategies provide an assuring method to incorporate the needed security measures. In this study, we have developed a complete prototype on Malicious URL Detection using machine learning methods. In particular, we have attempted an exact formulation of Malicious URL exposure from a machine learning perspective and proposed an approach using the AdaBoost algorithm—the proposed approach has brought forward more accuracy than other existing algorithms. The reason for choosing this particular boosting technique is that the AdaBoost algorithm can be used with any other machine learning algorithm.

**REFERENCES**


[1] J. Hong, "The state of phishing attacks," *Communications of the ACM*, vol. 55, no. 1, pp. 74–81, 2012.

[2] B. Liang, J. Huang, F. Liu, D. Wang, D. Dong, and Z. Liang, "Malicious web pages detection based on abnormal visibility recognition," in *E-Business and Information System Security*, 2009. EBISS'09. *International Conference on. IEEE*, pp. 1–5, 2009,.

[3] D. R. Patil and J. Patil, "Survey on malicious web pages detection techniques," *International Journal of u-and e-Service, Science and Technology*, vol. 8, no. 5, pp. 195–206, 2015.

[4] M. Cova, C. Kruegel, and G. Vigna, "Detection and analysis of drive-by- download attacks and malicious javascript code," in *Proceedings of the 19th international conference on World wide web*. ACM, pp. 281–290, 2010.

[5] R. Heartfield and G. Loukas, "A taxonomy of attacks and a survey of defence mechanisms for semantic social engineering attacks," *ACM Computing Surveys (CSUR)*, vol. 48, no. 3, p. 37, 2015.

[6] L. OpenDNS, "Phishtank: An anti-phishing site,", 2016, [Online], Available*:* https://www.phishtank.com.

[7] S. Sinha, M. Bailey, and F. Jahanian, "Shades of grey: On the effectiveness of reputation-based "blacklists"," in *Malicious and Unwanted Software, 2008. MALWARE 2008. 3rd International Conference on. IEEE*, pp. 57–64, 2008.

[8] S. Garera, N. Provos, M. Chew, and A. D. Rubin, "A framework for detection and measurement of phishing attacks," *in Proceedings of the 2007 ACM workshop on Recurring malcode*. ACM, pp. 1–8, 2007.

[9] S. Chhabra, A. Aggarwal, F. Benevenuto, and P. Kumaraguru, "Phi.sh/$ocial: the phishing landscape through short urls," in *Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*. ACM, pp. 92–101, 2011.

[10] Y. Alshboul, R. Nepali, and Y. Wang, "Detecting malicious short urls on twitter," 2015.

[11] Fossi, M., Egan, G., Haley, K., Johnson, E., Mack, T., Adams, T., and Wood, P., *Symantec internet security threat report*, trends for 2010. vol. 16, 2011.

[12] Prakash Pawan, Manish Kumar, Kompella Ramana, Gupta Minaxi, "PhishNet: Predictive Blacklisting to Detect Phishing Attacks," *INFOCOM, 2010 Proceedings IEEE*, pp. 1–5, 2010.

[13] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Identifying Suspicious URLs: An Application of Large-scale Online Learning," in *Proceedings of the 26th Annual International Conference on Machine Learning. ACM*, pp. 681–688, 2009.

[14] Ashutosh Marathe, Priya Jain, Vibha Vyas, "Iterative improved learning algorithm for petrographic image classification accuracy enhancement," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 1, pp. 289-296, Feb 2019.

[15] Komal KumarN, R. Lakshmi Tulasi, Vigneswari D., "An ensemble multi-model technique for predictingchronic kidney disease," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 2, pp. 1321-1326, Apr 2019.

[16] Y. Nakamura, S. Kanazawa, H. Inamura and O. Takahashi, "Classification of Unknown Web Sites Based on Yearly Changes of Distribution Information of Malicious IP Addresses," *2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*, Paris, pp. 1-4, 2018.

[17] G. Chakraborty and T. T. Lin, "A URL address aware classification of malicious websites for online security during web-surfing," *2017 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, Bhubaneswar, pp. 1-6, 2017.

[18] A. S. Manjeri, K. R, A. MNV and P. C. Nair, "A Machine Learning Approach for Detecting Malicious Websites using URL Features," *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*, Coimbatore, India, pp. 555-561, 2019.

[19] Manlin Wang, Fei Zhang and P. P. K. Chan, "Malicious website detection under the exploratory attack," *2013 International Conference on Machine Learning and Cybernetics*, Tianjin, pp. 565-570, 2013.

[20] S. B. Rathod and T. M. Pattewar, "A comparative performance evaluation of content based spam and malicious URL detection in E-mail," *2015 IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS)*, Bhubaneswar, pp. 49-54, 2015.

[21] P. J. Shinde and M. Chatterjee, "A Novel Approach for Classification and Detection of DOS Attacks," *2018 International Conference on Smart City and Emerging Technology (ICSCET)*, Mumbai, pp. 1-6, 2018.

[22] N. Kuze, S. Ishikura, T. Yagi, D. Chiba and M. Murata, "Crawler classification using ant-based clustering scheme," *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)*, London, pp. 84-89, 2015.

[23] S. D. Shirsat, "Demonstrating Different Phishing Attacks Using Fuzzy Logic," *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, Coimbatore, pp. 57-61, 2018.

[24] L. Biao, Z. Kejun, F. Huamin, Z. Kejun, F. Huamin and L. Yang, "A new approach of clustering malicious JavaScript," *2014 IEEE 5th International Conference on Software Engineering and Service Science*, Beijing, pp. 157-160, 2014.

[25] M. Nandhini and B. B. Das, "An assessment and methodology for fraud detection in online social network," *2016 Second International Conference on Science Technology Engineering and Management (ICONSTEM)*, Chennai, pp. 104-108, 2016.

## BIOGRAPHIES OF AUTHORS

**Firoz Khan** was born in Kerala, India, in 1974. He received a BSc degree in Electronics from the Bharatiyaar University, Coimbatore, India, in 1991 and a Masters Degree in information Technology from University of Southern Queensland, Australia, and another Master's Degree in Information Network and Computer Security (with Honors) from New York Institute of Technology, Abu Dhabi, UAE, in 2006 and 2016 respectively. He is currently working towards his PhD in Computer Science from the British University in Dubai, Dubai, UAE. In 2001, he joined the Higher Colleges of Technology in Computer Information Science department as a Teaching Technician continued on to become a Faculty member in 2005. He is currently holding the position of a Lecturer, with security and networking being his primary areas of teaching. His current research fields include computer security, machine learning, deep learning and computer networking.

**Mr. Jinesh Ahamed** was born in Kerala, India in 1979.He completed his bachelor degree in Physics from University of Calicut in 2000.He completed his Master's degree, MSc in computer communication and networking from Periyar university in 2002 and Masters in Network engineering from VIT University in 2005.He has been teaching in various international universities for the last 13 years. Currently he is working as Lecturer in CIS department in Higher college of technology teaching various courses in networking. Jinesh is pursuing his PhD in computer science from The British university in Dubai, UAE.His research areas include IoT technologies, security and artificial intelligence.

**Dr. Seifedine Kadry** has a Bachelor degree in applied mathematics in 1999 from Lebanese University, MS degree in computation in 2002 from Reims University (France) and EPFL (Lausanne), Ph.D. in applied statisticsin 2007 from Blaise Pascal University (France), HDR degree in 2017 from Rouen University. At present, hisresearch focuses on education using technology, system prognostics, stochastic systems, and probability andreliability analysis. He is an ABET program evaluator. He is an associate professor at the deparmtent ofMathematics and Computer Science, Beirut Arab University, Lebanon.

**Dr. Lakshmana Kumar Ramasamy** currently working as Assistant Professor cum Technical Trainer in Hindusthan College of Engineering and Technology, Coimbatore. Tamil Nadu. He is a global chapter Lead forMLCS [Machine Learning for Cyber Security]. Find him @ http://mlcsglobal.org/chapters-across-globe/. He iscurrently allied with company specific training of Infosys Campus Connect, Oracle WDP and Palo Altonetworks. He did his UG in PSG group of Institutions and completed his PG in Kalasalingam Universityand PhD under Anna University, Chennai and his research is on semantic web services. He has a passion towardsdevelopment and he holds the International certification on SCJP (Sun Certificated Java Programmer) and SCJWCD (Sun Certificate Java Web Component Developer). He is meticulous with programming languages likeJava, Python and PHP. He himself involves in research and expertise in distributed computing. He holds thecertification in Data Science from John Hopkins University, United States. He also holds the Amazon CloudArchitect certification from Amazon Web Services. He holds the privileged Gold level partnership award fromInfosys for bridging the gap between industry and academia in 2017.