e-ISSN 2320 –7876 www.ijfans.org
Vol.11, Iss.12, 2022

Detecting Malicious Websites Using Machine Learning

Mr. Shreyas Pagare, Dr. Manish Shrivastava, Dr. Manoj Verma

CSE Department, Chameli Devi group of Institutions, Indore (M.P.)

Abstract

In this study, we present the use of lexical characteristics, blacklists, DNS data, and web-based properties of domain names to identify dangerous domains. We will employ three well-known machine learning ensemble classifiers—Random Forest, Lite GBM, and XGBoost—using the attributes retrieved from the domain names to distinguish between benign and malicious domains. Active DNS analysis is the foundation of our experiment.

It is now feasible to foresee such websites, nevertheless, by employing machine learning algorithms on vast datasets. It is possible to identify harmful websites and alert visitors to the risk before they visit them using classifiers developed using methods like logistic regression and unsupervised machine learning. Using the Kaggle Malicious and Benign Website Dataset, this paper focuses on applying a range of unusual classification techniques to determine if a website is harmful or not.

Keyword: malicious domain detection, DNS, cyber security, malware, hishing, spam; domain generation.

Introduction

The World Wide Web as we know it is based on a variety of technologies, which makes choosing the very first website quite difficult. Yet we must begin somewhere, and where better to begin than with Tim Berners-Lee and his colleagues at CERN. They were in charge of developing the HyperText Transfer Protocol, which allowed servers and clients to connect, in 1989. There are an increasing amount of websites on the internet. 6.4% of the world's inhabitants, or 5.16 billion community, utilized the internet in January 2023, according to serve. of this total, 4.76 billion individuals, or 59.4 percent of the world's population, used social media.

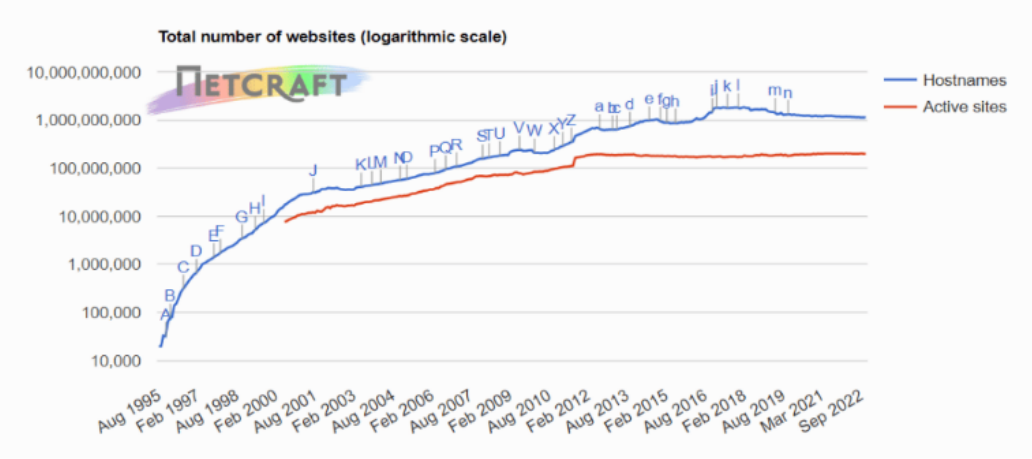

Figure 1: website (logarithmic scale)

The number of websites throughout time is seen in Figure 1 above. Yet as the internet grows, so does the chance of virus assaults on web services. With their websites, dishonest web developers distribute malware that may be used to attack servers and personal computers and violate privacy to commit fraud, theft, and blackmail.

The top 10 kinds of websites with dangerous content that may damage users are depicted in Figure 2 below. As can be seen, the categories listed below include some of the most popular websites that may be quite useful and ease the life of a user. This can turn into a user's worst nightmare when malevolent. These websites, which also provide retail, business, and gaming options, all ask visitors for their credit card numbers. The wrong persons might readily obtain this information, which could endanger the users' financial security.

For your reference, here are the top ten domains cited as being threatening to its visitors: Like .zip: file 100% evil and more than 1000 domains, Like .review: file 100% evil and more than 45304 domains, Like .country: file 99.9% evil and more than 5442 domains, Like .kim: file 99.74% evil and more than 8913 domains, Like .cricket: file 99.57% evil and more than 27723 domains, Like .science: file 99.35% evil and more than 324833 domains, Like .work: file 98.20% evil and more than 68144 domains , Like .party: file 98.20% evil and more than 206914 domains, Like .gq: file 97.68% evil and more than 69437 domains, Like .link: file 96.98% evil and more than 150595 domains.[21]

How can you maintain your company's network secure in the face of the numerous risks that exist online? The vast majority of harmful entities that can be discovered online cannot be mitigated by a standard firewall and antivirus program, and you cannot rely on your staff to spot spam and phishing schemes when they are needed.

Literature survey

The meaning of computer security

The majority of people's perceptions of netting security were based on the physical hardware prior to the concept of data security receiving extensive media coverage. Workstation security is the term used to describe the safeguarding of computing devices like laptops and smart phones, computer networks like private and public networks, as well as the entire Internet. Due to the ever-increasing confidence of most societies on computer systems, the field of information security, which includes all procedures and techniques that protect digital equipment, information, and services against accidental or illegal access, alteration, or destruction, is becoming more and more significant. It includes both information security, which protects the data kept on that device, and physical security, which stops equipment theft. Although it is occasionally referred to as cyber security or Information Technology security, these terms don't frequently apply to physical security.

However, there are evil actors online that register, weaponize, and use domains as part of spam, malware, or phishing schemes. The internet is less safe and more bothersome for everyone as a result of malicious websites. Prior to being weaponized and "crying 'Havoc!' and letting slip the dogs of war," these domains—domains registered with "malicious intent"—need to be identified and flagged.Based on a newly discovered domain dataset, Akamai researchers have classified nearly 79 million domains as harmful in the first half of 2022. 20.1% of all newly observed domains (NODs) that were successfully resolved are malicious domains, or over 13 million harmful domains per month.

In terms of coverage and mean time to detect, we compared a NOD-based detection technique with another well-known threat intelligence aggregator and discovered great complimentary value. We may examine the "long tail" of DNS requests using NOD-based threat detection, which enables us to identify brand-new malicious attacks extremely early in their lifespan.

The Domain Name System (DNS) is a multi-level addressing structure that sits on top of IP addresses. That enables material on a numeric address to be referred to by a name that is simple to remember, like cert.org. Also, it increases the number of naming possibilities to almost infinite

possibilities, such as hcjakaudbre.net or ajkcausdih.biz. Although some selections may be difficult to say, there are countless possibilities for addresses.

The Domain Name System (DNS) is organized into levels, which are groups of labels. The final element of the domain name is known as the first level or top level. www.google.com is an illustration of a three-level domain name, with a top-level domain of com, a succeeding-level domain of google, and a third-level domain of www. Hence, when your computer wants to search for www.google.com, it queries the domain name server root servers in charge of the.com TLD to determine which the authoritative DNS server for Google is based on the.com zone file. The answer would then be provided via the www.google.com numeric address provided by the google domain server. The same reverse resolution is used for addresses on other domains.

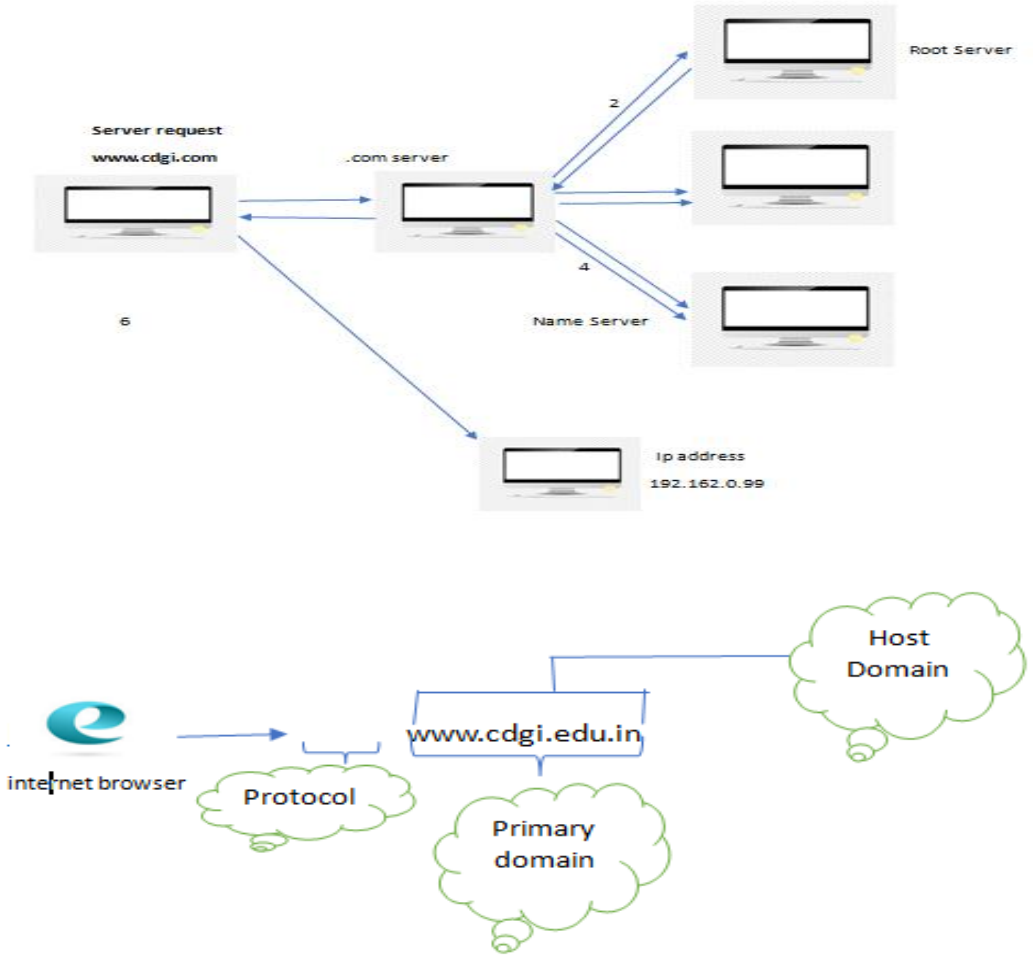

Figure 2: Working of Domain Name System (DNS)

Malicious Domains

A rogue website might expose your business to crimes like data theft or ransomware if an employee falls for its tricks. When a visitor performs an action, such clicking a link or installing software, malware is installed on the visitor's device in order to collect information. In other circumstances, no action is necessary, and anybody simply viewing the website might be infected with a "drive-by" download. These fraudulent websites recurrently pose as responsible ones and entice users with phishing emails. For instance, an employee could be asked to provide login information that might be used to access your business's network and steal crucial data. Alternatively, a worker can unintentionally download a piece of software or a file that could.

Cybercriminals create dangerous websites to disseminate ransomware and steal data. These sites typically assume legitimate identities while luring users with phishing emails. By security precautions and awareness training, employees can be kept from exposing themselves to risk and your company from suffering losses. Malware comes in many different forms, however the following ones are the most common:

Malicious Domain Detection

In comparison to static rules, machine-learning algorithms have been shown to be more successful in detecting DGA domains. The entropy, frequency of occurrence, top-level domain, number of dictionary terms, length of the domain, and n-gram, for instance, may be the input for various machine-learning techniques. Several of these methods, nevertheless, require tagged data. You must be familiar with several "good" domains and numerous DGA domains. DGA domains may be created using well-known malicious methods, whereas good domains can be obtained from sources like the Alexa and Majestic million sets. These DGA domains are legitimate, but only during the remaining period in which that particular method is used. There's a good likelihood that your model isn't compatible with any new DGAs. Unsupervised machine learning could provide a solution to the problems associated with tagged data. These methods merely need that you are aware of what is typical or anticipated; they do not require an explicit DGA training set. The majority of research focuses on neural network variations, which need a lot of computing power to train and forecast. If there is sufficient computational power, this is not always a deal-breaker with the amount of network data, but it is definitely something to take into account.

Related work

Many studies have been done in the previous that use various website features to assess whether malicious content is there. In (Chiba et al., 2012), the authors made the claim that IP address is a substantially supplementary reliable aspect of websites than URL and DNS and utilized machine learning to identify malware in websites using internet protocol address characteristics. Their research's objective was to create a potent technology to replace the existing ones. To categorise webpages, they deployed octet, extended octet, and bit string-based properties. Because domain names are inextricably linked to their fundamental communications, the DNS infrastructure, domain name analysis must be performed using domain name server data. To accomplish so, we have two approaches: one is active, in which we request domain name server data for a certain domain name, and the other is passive, in which we analyze real-time data at recursive domain name server servers or elsewhere on the domain name server infrastructure. We have limited ourselves to active domain name server analysis in this research. This section of the article lists a few of the most popular and effective studies in the fields of active and passive domain name server S analysis.

Khulood Al Messabi (2018) suggested a strategy for identify fraudulent domains based on domain name attributes and domain name server information [2]. They worked hard to discover and compile a list of dangerous TLDs and shocking terms that, when used in domain names, contribute to their maliciousness. They work on built a representation with the J48 decision tree classifier and 10-fold cross-validation.

To detect benign domain names, Chunyu Han and Yongzheng Zhang (2017) optional an come within reach of based on domain names and their TTL (time-to-live) attributes [15]. They employed the naive bayesian classifier method. They concentrated solely on establishing the benign-ness of a domain name, which may then be utilized to conclude the malicious-ness of a domain name.

Using website URL attributes such as textual qualities, link structures, webpage stuffing, domain name server information, and network interchange, up to date machine learning algorithms detect

dangerous websites" (Choi, Zhu and Lee, 2011). According to the authors, most long-established approaches target a single sort of harmful attack, however their methodology not only detects but also identifies the type of malicious assault. To detect malicious URLs, the authors employed an SVM, and to determine the sort of attack, they utilized a Multi-Label k-Nearest Neighbor technique. They completed each challenge with greater than 90% accuracy [16]. In summary, there have been a number of various ways tested in the literature. Each research has its own set of limitations, but one thing they all have in widespread is that they utilize rather primitive machine learning models and their characteristic set lacks demographic in sequence such as country of listing and registration dates.

Proposed model:

Regardless of whether a website is harmful or benign, a lot of data is linked with it. Previously, due to limited computer power and rudimentary analytics methods, it has been quite challenging to adequately examine this data. Yet, managing and analyzing vast amounts of data is now simple because to recent advancements in the disciplines of computer science, data analysis, and machine learning. In this paper, I will utilize supervised machine learning to determine if a website is harmful or benign using data that is frequently provided by websites all over the internet.

Machine Learning

A subfield of artificial intelligence called machine learning focuses on giving computers the capability to be taught from understanding and advance. The main goal is for the machine to be able to get data, use it, learn from it, and find pattern in it that can then be used to categorize, cluster, and make prediction Machine Learning is often divided into two categories: supervised machine learning and unsupervised machine learning[18]. Supervised Machine learning discovers patterns and connections between the input data and the result using a labeled set of examples. It then applies these insights to forecast the outcomes of fresh data [18]. Unmonitored With unlabeled data, machine learning attempts to reveal the hidden structure. Here, the emphasis is on inferring conclusions from the datasets rather than selecting the appropriate output[18] . So, we will be utilizing the three renowned machine learning ensemble classifiers Random Forest, Lite GBM, and XGBoost in this case study.

The Random Forest Classifier

A random forest, as its name suggest, is a collected works of different independent decision trees that function as an ensemble. The classification that obtain the nearly everyone votes defines the prediction generated by our model, which is generated by each tree in the random forest.

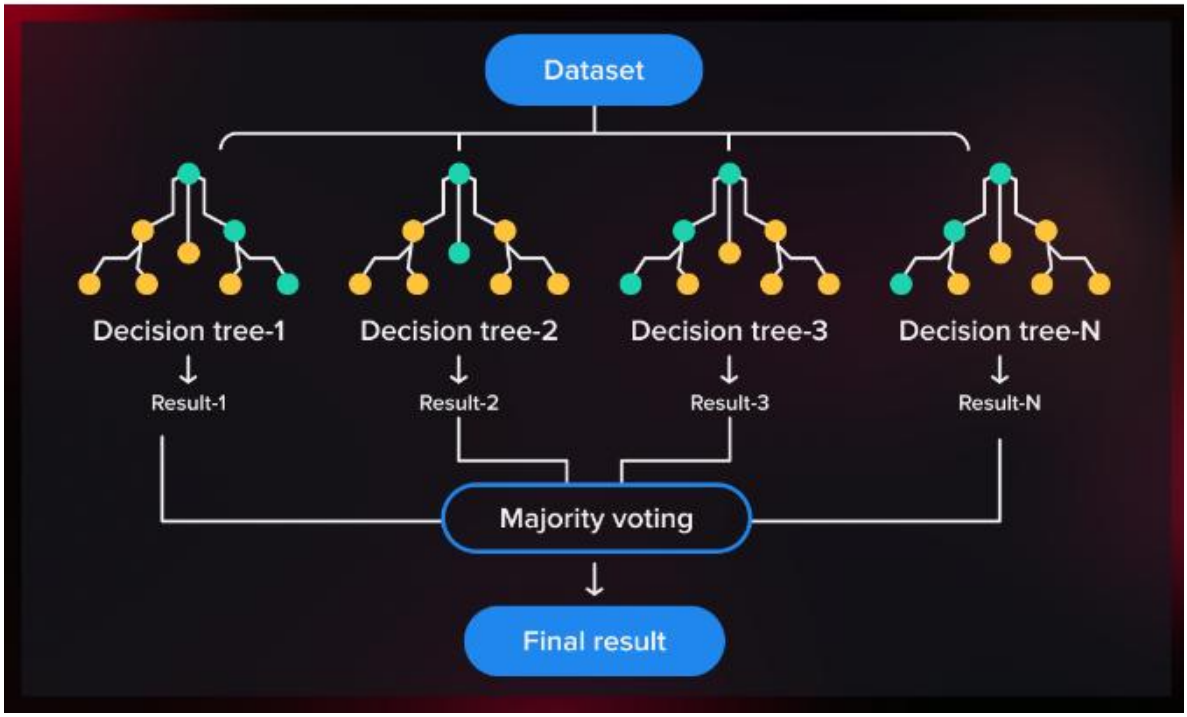

Figure 3: The Random Forest Classifier

The small connection between models is the key. Similarly to how assets with low correlations combine to produce a portfolio that is more than the sum of its parts, uncorrelated models can provide ensemble forecasts that are more truthful than any of the human being predictions. There must be some meaningful signal in those characteristics for the methods created using those characteristics to exceed guesses. Low correlation between the predictions made by the various trees are essential.

LightGBM, Light Gradient Boosting Machine

Microsoft has made its LightGBM tree-boosting library open source. According to their experiments, they outperform xgboost in terms of speed and memory.

This is the first time I've heard of this library, but this:

xgboost with max_depth=8 will have max number leaves to 255. This has same model complexity as LightGBM with num_leves=255, is a very misleading statement.

Reducing the depth of trees rather than the number of leaves will almost always result in lower performance than restricting the number of leaves because it stops the model from transferring capacity to areas of the domain where data density is high. It is not at all fair to compare depth-limited trees to leaf-limited trees.

Although the time results appear to be rather good, it is difficult to believe anything on that website because of the glaring error shown above. I'd be really curious to see an unbiased comparison of timings, which should be simple because it's free source, thank goodness. A fair comparison between this and xgboost would also interest me.

XGBoost for Applied Machine Learning

The XGBoost methodology has recently dominated Kaggle competition and applicable machine learning challenges for prearranged or tabular data. XGBoost is a gradient-boosted decision tree management solution for velocity and performance.

Generally, XGBoost is fast, that when compared to other implementations of gradient boosting its really fast.

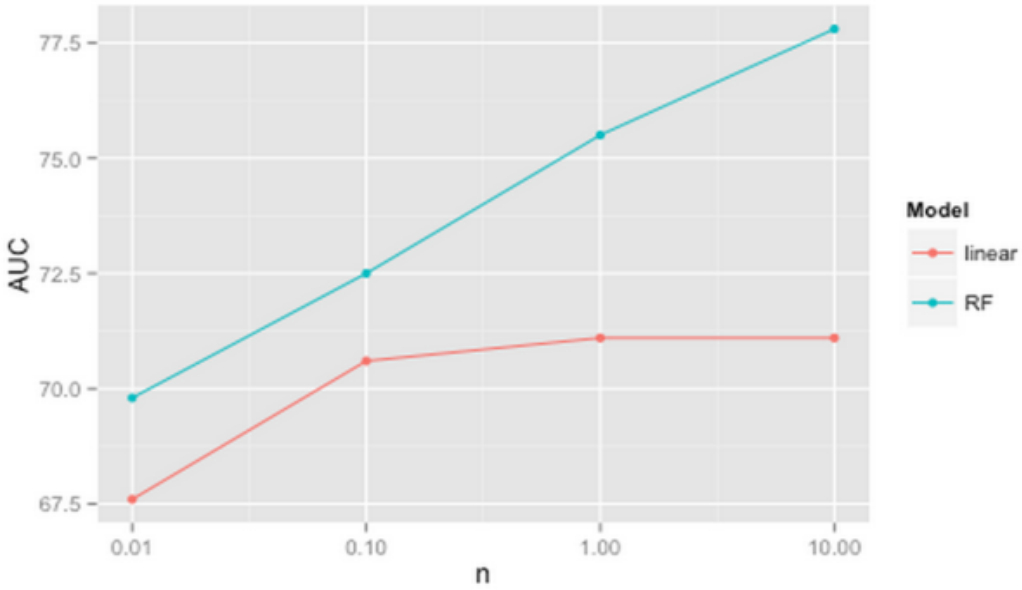

Figure 4: AUC

In this Paper I downloaded the Malicious URLs dataset from Kaggle to use for the data set. 651,191 URLs total, including 428103 benign or safe URLs, 96457 defacement URLs, 94111 phishing URLs, and 32520 malware URLs, have been gathered into a sizable dataset. For every project, selecting the right dataset for a machine learning project is one of the most significant household tasks. This dataset is a assemblage from five sources.

Wordcloud of URLs

It is one of the most interesting approach of natural language processing for analyzing the blueprint of word allotment. The word cloud of benign URLs in the next image makes this quite clear, with common tokens like html, com, org, wiki, etc. Phishing URLs frequently contain the tokens tools, ietf, www, index, battle, and net, although higher frequency tokens like html, org, and html are used because these URLs attempt to replicate the actual URLs in order to trick visitors.

In the word cloud of malware URLs, the tokens exe, E7, BB, and MOZI are more regular occurrences. These tokens are entirely visible because when a user visits a bad URL, the malicious URL tries to download executable files that are trojans onto the user's computer.

The alteration of tokens in its word cloud are more typical growth terminology like index, php, itemid, https, option, etc. because URLs' aim is to change the code of the original website.

Feature engineering: The following lexical characteristics will be extracted from raw URLs in this stage and utilized as input features to train the machine learning model. The ensuing characteristics are produced in the following ways:

Having ip address: To disguise the identity of a website, online criminals frequently use an Internet protocol address instead of the domain name server. This function will determine if the URL contains an IP address or not.

Abnormal Url: The WHOIS database may be used to extract this feature. Identity is commonly included in the URL of a trustworthy website.

Google index: Using this function, we determine whether or not the URL has been indexed by Google Search Console.

Count: The URLs of phishing or malware websites frequently contain more than two sub-domains. A dot separates each domain (.). Every URL with more than three dot characters (.) raises the risk of a malicious website.
Count-www: The majority of secure websites typically only contain one www in their URLs. If the URL has no www or more than one www, this feature aids in the detection of fraudulent websites.
Count@: If the URL contains the "@" sign, everything before it is ignored.
Count dir: Websites that have several directories in their URLs are typically suspect.
Count embed domain: Knowing how many embedded domains there are will help you spot dangerous URLs. You may accomplish it by looking for the character "//" in the URL.
URL has suspicious words: Suspicious terms like PayPal, login, sign in, banks, accounts, updates, bonuses, services, ebayisapi, tokens, etc. are frequently seen in malicious Websites. The existence of these frequently recurring suspicious terms in the URL has been identified as a binary changeable, i.e., whether such words are in attendance in the URL or not.
Short url: This feature tells you whether a URL has been shortened using a service, such as bit.ly, goo.gl, go2l.ink, etc.
Count https: Malicious Websites often avoid using HTTPS protocols since they typically demand user login information and guarantee that online transactions are secure. Hence, whether HTTPS is present or not is a key component of the URL.
Count http: Safe websites typically have a single HTTP in their URL, but phishing or malicious websites frequently have several HTTPs.
Count%: As we all know, spaces are not permitted in URLs. Normal URL encoding substitutes the symbol (%) for spaces. Secure websites typically have less spaces in their URLs than dangerous ones, which means that they have more spaces overall.
Count-: To make a Website appear legitimate, phishers and other cybercriminals frequently add dashes (-) to the brand name's prefix or suffix. An illustration. www.flipkart-india.com.
Count=: The equal sign (=) in the URL denotes that variables are being sent from one form page to another. As anybody may alter the values in a URL to change the page, it is regarded as being riskier.
Url length: To conceal the domain name, attackers frequently utilize lengthy URLs. We discovered that a safe URL typically has a length of 74 characters.
Hostname length: The hostname's length is a crucial element in identifying fraudulent URLs.
First directory length: With this feature, you may figure out how long the URL's first directory is. In order to get the first directory length of the URL, check for the initial '/' and count the length of the URL up to this point. Installing the Python TLD library is required to obtain directory-level information. You may install TLD by visiting this page.
Top-level domain length: one of the domains at the top of the Internet's hierarchical domain name system is a top-level domain (TLD). For instance, the top-level domain is com in the domain name www.example.com. So, the length of the TLD is crucial for recognizing fraudulent URLs. since.com is the most common extension for URLs. TLDs encompassing.
Count digits: Suspicious URLs are often those that contain numbers. Counting the amount of digits in a URL is a key characteristic for identifying fraudulent URLs because safe URLs often do not include digits.
Count letters: Another important factor in recognizing fraudulent URLs is the number of letters in the URL. Attackers typically accomplish this by adding more letters and numbers to the URL in an effort to lengthen it and conceal the domain name.

The dataset now appears as shown below following the creation of the 22 characteristics mentioned above.

df

	url	type	use_of_ip	abnormal_url	google_index	count.	count-www	count@	count_dir	count_embed_domian	...	count=
0	br-icloud.com.br	phishing	0	0	1	2	0	0	0	0	...	0
1	mp3raid.com/music/krizz_kaliko.html	benign	0	0	1	2	0	0	2	0	...	0
2	bopsecrets.org/rexroth/cr/1.htm	benign	0	0	1	2	0	0	3	0	...	0
3	http://www.garage-pirenne.be/index.php?option=...	defacement	0	1	1	3	1	0	1	0	...	4
4	http://adventure-nicaragua.net/index.php?optio...	defacement	0	1	1	2	0	0	1	0	...	3
...	...	...	...	...	...	...	...	...	...	...	...	...
651186	xbox360.ign.com/objects/850/850402.html	phishing	0	0	1	3	0	0	3	0	...	0
651187	games.teamxbox.com/xbox-360/1860/Dead-Space/	phishing	0	0	1	2	0	0	4	0	...	0
651188	www.gamespot.com/xbox360/action/deadspace/	phishing	0	0	1	2	1	0	4	0	...	0
651189	en.wikipedia.org/wiki/Dead_Space_(video_game)	phishing	0	0	1	2	0	0	2	0	...	0
651190	www.angelfire.com/goth/devilmaycrytonite/	phishing	0	0	1	2	1	0	3	0	...	0

651191 rows × 26 columns

Data Analysis

We will examine the distribution of several characteristics for each of the four types of URLs in this stage.

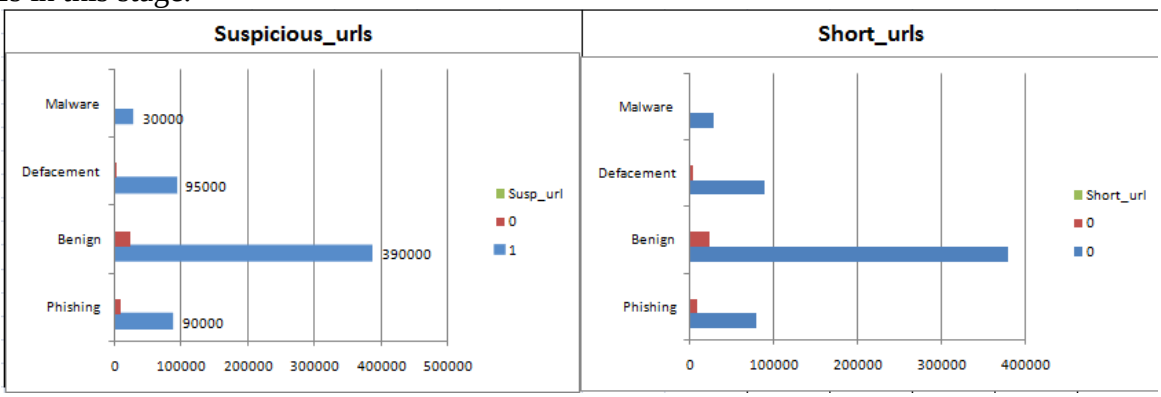

Figure 5: Data Analysis

Only malware Website includes IP addresses, as can be demonstrated from the distribution of the use ip address feature shown above. With abnormal urls, the distribution of defacement URLs is greater. According to the prevalence of suspicious urls, phishing URLs have the second-highest distribution, followed by benign URLs, which have the greatest distribution. While real banking or payment-related URLs typically contain these keywords, suspicious URLs also include them, which explain why benign URLs have the largest distribution. Since we know that most of the time we utilize URL shortening services to make it easier to share large URLs, we can see from the short url distribution that benign URLs have the highest percentage of short URLs.


```
df=pd.read_csv('/content/malicious_phish.csv')
print(df.shape)
df.head()
```

```
(651191, 2)
```

	url	type
0	br-icloud.com.br	phishing
1	mp3raid.com/music/krizz_kaliko.html	benign
2	bopsecrets.org/rexroth/cr/1.htm	benign
3	http://www.garage-pirenne.be/index.php?option=...	defacement
4	http://adventure-nicaragua.net/index.php?optio...	defacement

Model evaluation & comparison

On the test set, we have predicted. This is a performance comparison between Lite GBM, XGBoost, and Random Forest. As demonstrated by the outcomes listed above, Random Forest performs best in terms of test accuracy, achieving the greatest accurateness of 96.6% and having a better discovery rate for malware, phishing, and benign content. We have chosen Random Forest as our primary model for identify malicious URLs based on the presentation describe above, and in the next phase, we will also plot the characteristic significance plot.

Random Forest

	Precision	Recall	f1-score	support
benign	0.97	0.98	0.98	85621
defacement	0.98	0.99	0.99	19292
phishing	0.98	0.94	0.96	6504
malware	0.91	0.86	0.88	18822
accuracy			0.97	130239
macro avg	0.96	0.95	0.95	130239
weighted avg	0.97	0.97	0.97	130239
accuracy:	0.966			

XG Boost

	Precision	Recall	f1-score	support
benign	0.97	0.99	0.98	85621
defacement	0.97	0.99	0.98	19292
phishing	0.98	0.92	0.95	6504
malware	0.91	0.83	0.87	18822
accuracy			0.96	130239
macro avg	0.96	0.93	0.94	130239
weighted avg	0.96	0.96	0.96	130239
accuracy:	0.962			

Light GBM

	Precision	Recall	f1-score	support
benign	0.97	0.99	0.98	85621
defacement	0.97	0.99	0.98	19292
phishing	0.98	0.92	0.95	6504
malware	0.91	0.83	0.86	18822
accuracy			0.96	130239
macro avg	0.95	0.93	0.94	130239
weighted avg	0.96	0.96	0.96	130239
accuracy:	0.95			

We will check highly contributing features in Random Forest next. The code used to plot the relevance of features. The top 5 characteristics for identifying malicious URLs, according to the figure above, are hostname length, count dir, count-www, fd length, and url length.

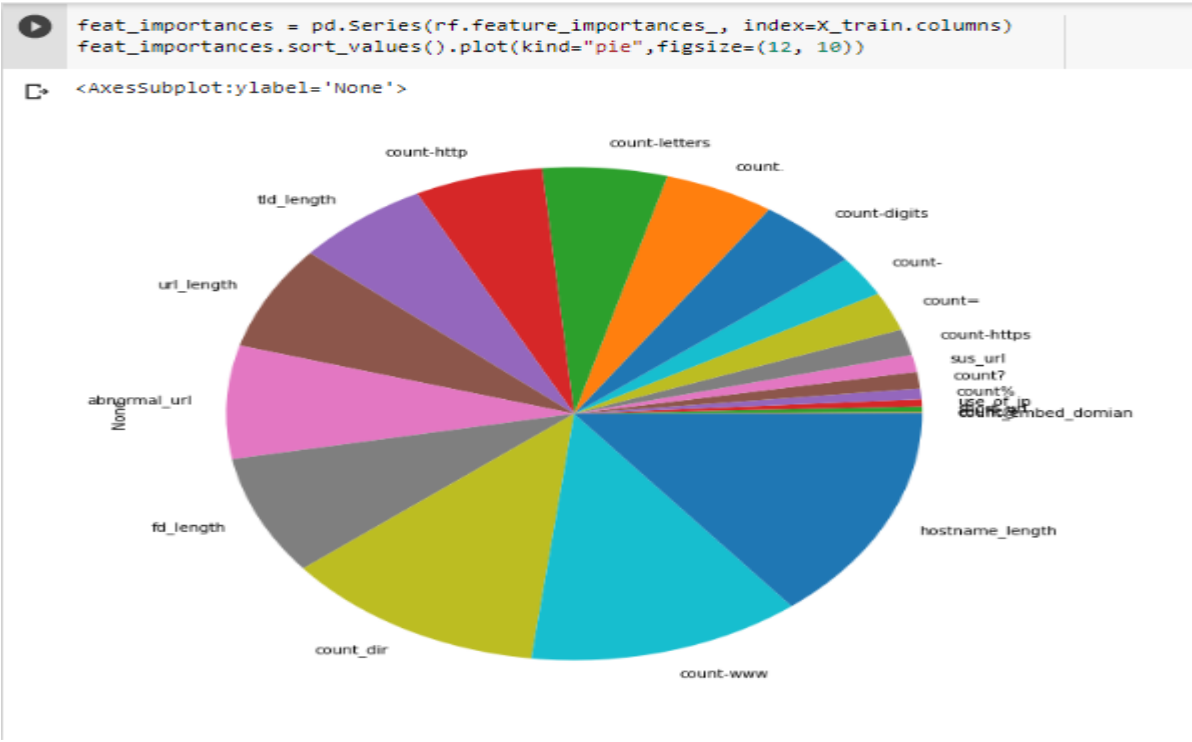

Figure 6: Data Analysis Graph

Conclusion

Using raw URLs, we extracted 22 lexical characteristics and trained the XG Boost, Lite GBM, and Random forest machine learning models. In addition, we examined the efficacy of the three machine learning models and discovered that Random forest performed superior than the others, achieving the maximum correctness of 96.6%. We discovered that the top 5 features for identify malicious URLs are hostname length, count dir, count-www, fd length, and url length by charting the feature importance of Random forest. The prediction method for categorization any raw URL using our stored model, Random Forest, has finally been written.

Reference

1. Oladimeji, S. S. M. K. (2021, June 16). SolarWinds hack explained: Everything you need to know. WhatIs.Com. https://whatis.techtarget.com/feature/SolarWinds-hack-explained-Everything-you-need-to-know
2. Srivastava, S. (2020, September 8). International Literacy Day: Bridging India's Digital Divide. BloombergQuin.https://www.bloombergquint.com/technology/international-literacy-day-bridging-indiasdigitalivide#:%
3. D.A.Y. (2020, December 18). Cyber Security: The Emerging Need of India. Wesleyan Journal of Research, 13.
4. Team—Dcn, R. (2016, February 11). Sixty percent of mobile banner clicks are accidental. Digital Content Next. https://digitalcontentnext.org/blog/2016/02/10/sixty-percent-of-mobile-banner-clicks-are-accidental/
5. Damiani, J. (2020, March 26). Google Data Reveals 350% Surge In Phishing Websites During Coronavirus andemic.Forbes.https://www.forbes.com/sites/jessedamiani/2020/03/26/google-data-reveals-350-surge-in-phishing-websites-during-coronaviruspandemic/?sh=45f4cbac19d5
6. D.A.Y. (2021). PHISHING ATTACKS IN INDIA: ISSUES AND CHALLENGES. Sambodhi (UGC Care Journal), 44(1), 27–31.

7. Dr. Anusuya Yadav. (2021). Cyber-Crime: A Study of Gurugram and Rohtak Districts. International Journal of Modern Agriculture,10(1), 942 - 948. Retrieved from http://www.modern-journals.com/index.php/ijma/article/view/696
8. Joshi, M. C. A. (2019, May 23). Phishing in India is becoming innovative. Indiaforensic. https://indiaforensic.com/understandingphishing-india/
9. Shubhangi Taneja; Ruchi Pal; Shiwangi Vishwakarma; Rakesh Kumar. "A Case Study on Cyber bullying". International Research Journal on Advanced Science Hub, 2, 7, 2020, 29-31. doi: 10.47392/irjash.2020.60
10. C. Choudhary, R. Sivaguru, M. Pereira, B. Yu, A. C. Nascimento, M. De Cock. 2018. "Algorithmically generated domain detection and malware family classification", Proceedings of the Sixth International Symposium on Security in Computing and Communications, 2018.
11. M. Antonakakis, R. Perdisci, W. Lee, N. Vasiloglou, and D. Dagon. 2011. Detecting malware domains in the upper DNS hierarchy. In the Proceedings of 20th USENIX Security Symposium (USENIX Security '11), 2011.
12. Shuang Hao, Matthew Thomas, Vern Paxson, Nick Feamster, Christian Kreibich, Chris Grier, and Scott Hollenbeck. 2013. Understanding the domain registration behavior of spammers. In Proceedings of the 2013 conference on Internet measurement conference (IMC '13). ACM, New York, NY, USA, 63-76.
13. ANDYEN T, REITER M. 2008. Traffic aggregation for malware detection. In Conference on Detection of Intrusions and Malware& Vulnerability Assessment (DIMVA) (2008).
14. Eshete, Birhanu & Villafiorita, Adolfo & Weldemariam, Komminist. 2013. BINSPECT: Holistic analysis and detection of malicious web pages. SecureComm. 2012.
15. Han, Chunyu & Zhang, Yongzheng. (2017). CLEAN: An approach for detecting benign domain names based on passive DNS traffic, in Proceedings of 6th International Conference on Computer Science and Network Technology (ICCSNT) 343-346.
16. Ashishsubedi, 2020. 5 Machine Learning Algorithms for Beginners. [online] Medium. Available at: <https://medium.com/analytics-vidhya/5-machine-learning-algorithms-for-beginners-67e8b95d1e2a> [Accessed 9 Apr. 2020].
17. Breiman, L., 2001. Random Forests. Machine Learning, 45(1), pp.5–32.
18. Brown, J.B., 2018. Classifiers and their Metrics Quantified. Molecular Informatics,[online]37(12).Availableat:<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5838539/> [Accessed 6 Apr. 2020].
19. Chiba, D., Tobe, K., Mori, T. and Goto, S., 2012. Detecting malicious websites by learning IP address features. In: Proceedings - 2012 IEEE/IPSJ 12th International Symposium on Applications and the Internet, SAINT 2012. pp.29–39.
20. Choi, H., Zhu, B.B. and Lee, H., 2011. Detecting Malicious Web Links and Identifying Their Attack Types.
21. Eshete, B., Villafiorita, A. and Weldemariam, K., 2011. Malicious website detection: Effectiveness and efficiency issues. In: Proceedings - 1st SysSec Workshop, SysSec 2011. pp.123–126.
22. Fawagreh, K., Gaber, M.M. and Elyan, E., 2014. Random forests: from early developments to recent advancements. Systems Science & Control Engineering, 2(1), pp.602–609.
23. https://www.xfer.com