Phishing URL Detection

Author: **Prof. Sneha Bombe, Mr. Aditya Sao, Mr. Mukul Zade, Mr. Mohit Kadre, Mr. Piyush Dodke, Mr. Vaibhav Sakharwade**.
Jhulelal Institute of Technology,Village Lonara, Off, Koradi Rd,
Nagpur, Maharashtra 441111

E-Mail ID: admin@jitnagpur.edu.in

***Abstract*— Phishing remains one of the simplest yet most dangerous techniques used by cybercriminals to steal confidential data from unsuspecting users. Attackers attempt to deceive individuals into disclosing vital details such as logon credentials, bank information, or personal identification numbers. Cybersecurity experts continue to develop dependable systems to accurately differentiate between legitimate and fraudulent websites. This study applies machine learning models to examine various URL characteristics in order to detect phishing attempts. Models Like 'The Support Vector Machine' (SVM), 'The Decision Tree', and 'The Random Forest' are trained and compared based on performance indicators including validity, false +ve rate, and false negative rate. The main goal of this research is to acquire the most effectual algorithm for phishing URL detection through detailed comparative analysis.**

***Keywords*— Phishing, URL classification, machine learning, URL features, website detection, data security**

I. INTRODUCTION

The rapid expansion of the internet has made phishing one of the most common forms of cyber deception. Creating a fake web page that looks almost identical to a genuine one is relatively easy. While cybersecurity specialists can usually spot such fakes, ordinary users often fail to recognize them, resulting in serious data breaches. The primary motivation behind phishing attacks is to gain access to sensitive financial information such as online banking credentials. It is estimated that business market in the United States of America lose approximately $2 billion annually due to phishing-related incidents. Furthermore, a report by Microsoft in 2014 estimated that phishing might cost the global economy nearly $5 billion. The growing success of phishing stems from users' lack of awareness and the psychological manipulation used by attackers

Traditional detection methods, such as blacklists, rely on databases containing previously identified malicious URLs. However, attackers frequently evade these lists by constantly modifying their domain names or employing fast-flux techniques that rapidly switch hosting servers. These tactics enable them to launch "zero-hour" attacks that blacklists fail to detect.

Although "heuristic- detection", which incorporates traits that have been seen to be present in "actual phishing", is capable of identifying zero-hour phishing attempts, the false positive rate in detection is quite high and the "traits are not always present in such attacks".
To solve the shortcomings of blacklist and heuristics based strategy, many IT security experts are currently concentrating on machine learning solutions. Machine learning technology contains numerous algorithms which needs previous data to make a conclusion or prediction of future data..

III. DATASET

The dataset used for this investagaton consists of URLs collected from verified sources. Legitimate website links were combined with phishing URLs retrieved from www.phishtank.com. The dataset contains a total of 36,711 entries, with 17,058 legitimate and 19,653 phishing URLs. Each entry is labeled as "1" for phishing and "0" for benign.

III. FEATURE EXTRACTION

1. **IP-Based Domains:** Most trustworthy websites map their services to domain names. URLs containing raw IP values are frequently used in temporary or malicious hosting environments**.**
2. **Use of the "@" Character:** Including this symbol can cause browsers to ignore preceding text, enabling misleading redirections.
3. **Excessive Dots:** When the hostname contains more than three dot levels, the URL may be intentionally segmented to mimic wellknown brands.
4. **Hyphenated Domains:** Attackers commonly insert dashes between brand-related phrases to produce look-alike addresses**.**
5. **Multiple Redirections:** Additional "//" patterns within the URL may indicate layered forwarding to suspicious destinations
6. **Misplaced Security Tokens**: Words associated with encryption, such as "HTTPS," used within the domain itself are sometimes used to create a false sense of legitimacy.
7. **Email-Handling Keywords:** Terms like mailto: or usage of mail() scripts are unusual for ordinary login pages, suggesting malicious data capture.
8. **Shortened Links:** URL shortening services obscure true destinations and are exploited to conceal phishing endpoints.
9. The avg length of benign URLs is 25. If a URL is larger than 25, the feature is set to 1; otherwise, it is set to 0.
10. **Abnormally Long URLs:** Genuine websites typically keep addresses concise. Unnecessarily lengthy URLs often hide querystring manipulation.
11. **Sensitive Lexical Terms:** Words such as "verify," "reset," "secure," and "update" attempt to trigger urgency in users.
12. **High Slash Count:** Too many slash-separated directories may represent indirect forwarding steps.
13. **Unicode Domain Tricks:** Certain Unicode character sets visually resemble Latin letters, enabling impersonation.
14. **SSL Certificate Duration:** Reputable services usually maintain certificates for extended periods rather than short-term issuance.
15. **External Anchor Behavior:** Pages whose hyperlinks consistently point to unrelated domains may be harvesting user data.
16. **Invisible IFrames:** Hidden frames can overlay malicious input fields or fraudulent content.
17. **Low Website Rank:** Unknown or rarely visited domains are statistically more likely to host phishing resources.

Domain Registration Length

Phishing pages are commonly hosted on domains that have been registered for a very short duration. Attackers create these domains only temporarily to avoid tracking and blacklisting. Domains with a longer registration period are more likely to be owned by reputable organizations**.**

Presence of Suspicious Ports

Standard websites typically operate on well-known ports such as 80 (HTTP) and 443 (HTTPS). When a URL explicitly specifies non-standard ports (e.g., :8080 or :81), it may indicate unauthorized hosting or experimental servers. Many phishing campaigns exploit random ports to bypass access control.

Favicon Consistency

The favicon is a small website icon displayed in the browser tab. If the favicon is loaded from an external or unrelated source, it can suggest that the page is imitating another site. Legitimate platforms usually store their favicon within their own domain for branding consistency.

JavaScript Obfuscation

Phishing sites often include obfuscated JavaScript to hide code that steals credentials. Obfuscation patterns, such as encoded characters or unreadable variable names, are red flags. Legitimate sites use minimal obfuscation, primarily for optimization and protection of intellectual property.

Status-Bar Customization

Some malicious webpages modify or disable the browser's status bar using JavaScript events. By hiding true hyperlink destinations, attackers mislead users about where they are navigating. This behavior is rarely found in trustworthy websites.

URL Host Length

If the hostname portion of a URL is unusually long, it may be intentionally padded with keywords or random strings to confuse users. Attackers frequently use this tactic to mimic popular domain names. Legitimate hosts typically follow concise naming conventions.\

Redirection Count
Multiple redirections occur when a user is forwarded through several intermediary URLs before reaching the final page. While sometimes used legitimately for tracking, excessive redirection chains are common in phishing infrastructures. This helps criminals hide their true hosting environment.

Subdomain Repetition
Attackers may append multiple subdomains to imitate credible websites such as secure.login.account.bank.com.evil123.com. Repetitive subdomains can deceive users at a glance. Well-established services rarely rely on long hierarchical subdomain chains.

Content Delivery from Mixed Sources
If a webpage loads scripts, images, or fonts from many unrelated external domains, it indicates dependency on unknown servers. Phishing attackers often borrow resources from various hosts to assemble fake pages quickly. Legitimate platforms usually minimize external dependencies for security.

IV. MACHINE LEARNING ALGORITHM

In this Project we Are Implementing Three ML The Grading model 'Random Forest' 'Decision Tree', and 'Support Vector Machine' has been used to detect phishing website.

4.1 Decision Tree Algorithm

A Decision Tree classifies data by repeatedly partitioning samples based on the attribute that best separates categories. Each branching point represents a conditional test, and the final leaves correspond to predicted labels. Since the logic is expressed visually, decision boundaries are easy for analysts to interpret, making this approach useful for binary classification tasks such as phishing detection.

4.2 Random Forest Algorithm

Random Forest models build many Decision Trees by using randomly chosen subsets of both features and data. Each tree contributes a vote toward the final classification, reducing the risk of overfitting found in individual trees. The ensemble structure typically results in improved robustness and accuracy

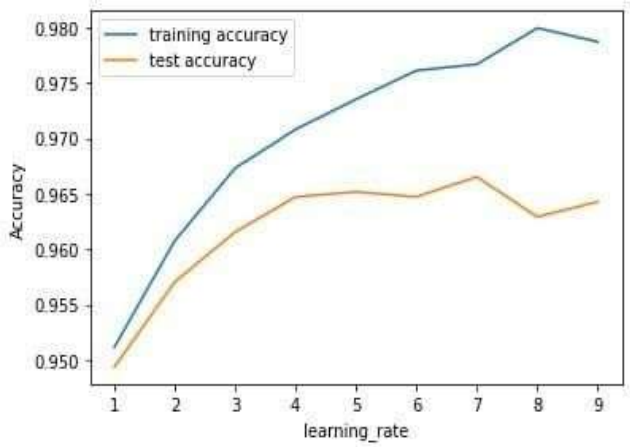
Tree construction is based on the bootstrap method., which develops one tree by random sampling with replacement of dataset features and observations. Also, internally of random forest, the best division for classification is chosen among a random subset of the features, relying on impurity measures such as the Gini index and information gain. This is in analogy with decision tree algorithms.. This process is repeated until a forest comprising n trees is obtained.

4.3 Support Vector Machine Algorithm [7]

SVMs aim to find a separating boundary between classes that maximizes the distance to the closest points from each class. These “support vectors” influence where the boundary forms. When data is not linearly separable, kernel functions map samples into more complex feature spaces, enabling the algorithm to identify effective boundaries in difficult problems.

A support vector machine now creates a dividing line that is perpendicular to the join line. To classify the data correctly, the margin of error needs to be as large as possible. In this case, the margin refers to the distance between the hyperplane and the support vectors.“clear separation in the data space using a **hyperplane”**. In practice, though, we often deal with messy, non-linear data that can't be separated by a simple line. To overcome this limitation, SVMs utilize **kernel tricks**, a method that transforms the data into a higher-dimensional format, allowing the algorithm to find that optimal linear boundary even in complex scenarios.

4.4 Gradient boosting Imagine building a strong team by adding members one by one. **Gradient boosting** does this by simple prediction models . Each new model joins the team to fix the mistakes the previous models made. The system uses a method called **gradient descent** to figure out exactly how to correct those errors, gradually getting more accurate. It also uses a "learning rate" (shrinkage) to ensure each new member contributes gently, which prevents the team from becoming too specialized and ensures they work well on new challenges.



and subsampling, makes gradient boosting a highly effective and adaptable method for both classification and regression tasks.

V. IMPLEMENTATION AND RESULT

Nowadays, we frequently utilize the Scikit-learn library to access machine learning algorithms. The dataset is divided into training and testing subsets with varying ratios: 49:51, 70:30, and 90:10. Every classifier is trained with the training set, while the testing set evaluates the classifiers' performance. We determine their performance by computing the accuracy score, false positive rate, and false negative rate

Dataset Split ratio	**Classifiers**	**Accuracy Score**	**False Negative Rate**	**False Positive Rate**
50:50	Decision Tree	93.18	3.44	3.12
	Random Forest	98.72	4.69	2.91
	Support vector machine	95.87	4.99	3.01
70:30	Decision Tree	92.76	2.43	3.11
	Random Forest	97.00	4.23	2.56
	SVM The support vector machine	97.82	4.19	1.99
90:10	Decision Tree	96.22	2.99	2.67
	SVM The support vector machine	**98.12**	**2.98**	2.11
	Random Forest	95.21	5.00	1.34

VI. CONCLUSION

In this paper, we try to improve the detection method for phishing sites by implementing machine learning technology. We were able to reach the lowest false positive rate and a high detection accuracy of 93.72% by implementing the 'random forest algorithm'. While all this we gain, the results that indicate that classifiers accomplish better when we use more test training data.

The hybrid technology in future will be able to make more precise identification of phishing websites by using the random forest algorithm of machine learning technology and the blacklist method.

REFERENCES

[1] Gunter Ollmann, “The Phishing Guide Understanding & Preventing Phishing Attacks”, IBMInternet Security Systems, 2007

[2] **KDnuggets – Support Vector Machines Explanation**.
Available:
https://www.kdnuggets.com/2016/07/supportvectormachines-simple-explanation.html

[3] **Data Aspirant – Decision Tree Algorithm**.
Available:
http://dataaspirant.com/2017/01/30/how-decisiontreealgorithm-works/