International Journal of Scientific Research and Management (IJSRM)
||Volume||12||Issue||04||Pages||484-493||2024||
Website: https://ijsrm.net ISSN (e): 2321-3418
DOI: 10.18535/ijsrm/v12i04.m02

Impact of Web (URL) Phishing and Its Detection

Kunle Oloyede[1], Chinenye Obunadike[2], Simo Yufenyuy[3], Emmanuel Elom[4], Abdul-Waliyyu Bello[5], Somtobe Olisah[6], Callistus Obunadike[7], Oluwadamilola Ogunleye[8], Sulaimon Adeniji[9]

[1, 3, 4, 5, 6,7] Department of Computer Science and Quantitative Methods, Austin Peay State University, Clarksville, USA
[2] Anambra State University Uli, Anambra State Nigeria
[8] George Washington University, Washington DC, USA
[9] University of Lagos, Lagos State Nigeria

Abstract:
Web phishing is a persuasive and evolving cyber threat that poses significant risks to individuals, businesses, and organizations in the modern digital age. This paper aims to provide an overview of web phishing, focusing on its methods, detection techniques, and prevention. Phishing attacks occur when malicious actors use deceptive practices to trick people into divulging sensitive or classified information such as passwords, credit card details, or personal data. These attacks primarily manifest through emails, websites, or social engineering tactics. Phishing emails often impersonate trusted entities and lure recipients into clicking on malicious links or downloading harmful attachments. Web phishing involves using fraudulent websites that mimic legitimate ones to steal user information or deliver malware. Detecting web phishing attacks is an ongoing challenge due to the sophistication of attackers. Several detection techniques have been developed, including heuristic analysis, machine learning algorithms, and real-time URL analysis. These methods analyze various attributes of websites and emails to identify suspicious patterns or behaviors. Detecting web phishing is equally essential. Effective prevention strategies include user education and awareness programs, using two-factor authentication, regular software updates, and deploying advanced email filtering and anti-phishing tools. User training is crucial in helping individuals recognize phishing attempts and avoid falling victim to them

Keywords: Web phishing*, machine learning, real time URL analysis detection techniques*

1. Introduction

The primary objective of this paper is to effectively categorize and address the menace of phishing web pages, specifically URLs, which can severely jeopardize individuals' social and financial well-being. To embark on this endeavor, gaining a comprehensive understanding of the phishing phenomenon and the significance of web security within this context is crucial. Phishing is a cybercrime wherein malicious actors aim to get sensitive information from unsuspecting users illicitly [1]. These perpetrators fabricate deceitful and cruel online platforms, complete with counterfeit web addresses. Subsequently, their nefarious intentions manifest as they employ tactics to manipulate unsuspecting targets into interacting with these malicious web links or landing on deceptive web pages [2]. Once enticed into accessing such fraudulent URLs, users find themselves prompted to divulge their personal information [3]. This data is then exploited for various illicit purposes, including potential unauthorized access to the victim's bank accounts [1]. This document provides a comprehensive overview of the strategies employed to combat web-based phishing attacks, introducing a model designed to identify and categorize phishing URLs effectively. By emphasizing the importance of recognizing and addressing the threat of phishing, this paper aims to enhance online security and protect individuals' sensitive information.

Phishing: Phishing is a cybercrime where malicious actors attempt to acquire an individual's private information illicitly, often including sensitive data like bank accounts or social media credentials, to deceive the victim. Unsuspecting users inadvertently divulge their confidential information when they access a specific URL, potentially resulting in severe consequences for the individual [4].

Classification: Classification is a supervised technique of learning in data mining. The classification technique first learns the predefined patterns and can then identify the trained patterns.
URL classification: Most of the time, in sorting, the data patterns are available in a structured manner. However, the URL data is not available in a fixed design. Hence, apply classification or machine learning techniques in URL data [5].

Phishing Detection: Phishing is a crucial issue in web security. The phishing detection techniques enable us to identify the phishing URLs by evaluating the URLs. To assess the URLs, several methods are available, i.e., blacklist and whitelist-based techniques, statistical analysis-based techniques, and machine learning-based techniques [6]. Amongst the available methods, machine learning techniques are more efficacious and accurate. In such scenarios, the malicious URL patterns are learned by classification algorithms, and when required, the URL types (phishing or legitimate) [7].

Malicious URLs: The URLs that contain fake information or duplicate web page of any authentic and reputed web page that act or deviate from the actual URL behavior is termed malicious or phishing URL [8].

Phish Tank Database: The phish tank database is a standard repository that records phishing-reported URLs by different web security agencies. This database contains several additional attributes between essential details (features) listed as Reporting date, Phishing target, URL, Reported agency, and others.

1.1 How Web Phishing Typically Works:

1.1.1 Deceptive Emails: Attackers send misleading emails that appear to come from legitimate sources, like banks, social media sites, or trusted companies [9]. These emails often contain urgent messages or enticing offers to lure recipients into action.

1.1.2 Fake Websites: These emails may contain links that lead to counterfeit websites that closely resemble legitimate ones. These websites trick users into thinking they're on the actual site.

1.1.3 Information Collection: Once on the fake website, users are prompted to enter sensitive information such as usernames, passwords, credit card numbers, or other personal details. The attackers then capture this information.

1.1.4 Malware Distribution: Some phishing attacks may also involve the distribution of malware. Clicking on a malicious link or downloading an attachment from a phishing email can infect the user's device with malware, which can then steal data or take control of the device [10].

1.2. Consider The Following Tips To Protect Yourself From Web Phishing Attacks:

1.2.1 Skeptical: Always be cautious when receiving unsolicited emails, especially those with urgent requests or offers that seem too good to be true.

1.2.2 Verify the Sender: Check the sender's email address and verify it matches the legitimate source's official domain.

1.2.3 Hover Over Links: Before clicking on any link in an email, hover your mouse over it to see where it leads. Ensure it goes to a legitimate website.

1.2.4 Two-Factor Authentication (2fa): Enable 2FA wherever possible, especially for sensitive accounts like banking and email. This technology adds an extra layer of security [11].

1.2.5 Keep Software Updated: Ensure your operating system, browser, and security software are up to date to protect against known vulnerabilities.

1.2.6 Educate Yourself: Familiarize yourself and your employees (if applicable) with phishing tactics and how to recognize them. Regular training can be very effective in preventing attacks.

1.2.7 Report Suspected Phishing: If you receive a suspected phishing email, report it to your email provider and the Anti-Phishing Working Group (APWG) at reportphishing@apwg.org.

1.2.8 Antivirus/Malware Scanner: Use reputable antivirus and anti-malware software to scan and protect your devices from malicious software. Remember that phishing attacks can be highly sophisticated, and attackers constantly evolve tactics [12].

2. 2. Literature Reviews

This section encompasses a range of contemporary contributions and research papers dedicated to enhancing conventional approaches to phishing categorization. There's a growing demand for automated solutions to identify and mitigate the rising volume of malicious content on social media [13]. Specifically, [14] laid the groundwork for a superior machine learning classification model in 2018, orchestrated by the Anti-Phishing Working Group (APWG). Their initiative aimed to pinpoint the proliferation of harmful content within online social networks and media, leveraging a dataset encompassing 51,401 unique phishing websites from the abovementioned list. This endeavor primarily focused on identifying social network posts featuring inflammatory language and Uniform Resource Locators (URLs) leading users to sites housing malicious material, encompassing drive-by downloads, phishing schemes, spam, and scams. The dataset underwent labeling through Virus Total assistance, utilizing the Twitter streaming API for data collection. Fraudulent emailing, a criminal tactic designed to extract sensitive user information, including personal data, login credentials, and other confidential details, remains a big concern. Phishing, characterized by acquiring sensitive consumer information like passwords, bank account details, credit card numbers, financial credentials, etc., is a process that nefarious actors may exploit later [15]. The primary objective of this paper is to identify page similarities in basic visual elements constituting the appearance of web pages. The proposed system introduces a combination of the Support Vector Machine (SVM) approach, spam map reduction, and image spam filtering to enhance the accuracy of spam URLs and image spam recognition. Over the past decade, many counterfeit websites have surfaced on the World Wide Web, designed to resemble reputable platforms to deceive consumers and businesses, aiming to misappropriate funds. As an online attack vector, phishing has resulted in considerable financial losses for the online community and stakeholders, amounting to hundreds of millions of dollars. Effectively countering phishing necessitates the implementation of robust countermeasures. In contrast to traditional anti-phishing strategies like awareness workshops, visualization techniques, and legal remedies, machine learning, a popular method for data analysis, has recently displayed promising results in the battle against phishing [16].

Experimental findings underscore that coverage-based models are particularly suitable as anti-phishing solutions for novice users due to their high phishing detection rates. [17] delve into deploying data mining algorithms, specifically categorization and association algorithms, underpinning a clever and successful model for identifying phishing websites, and the connections that link them. These algorithms uncover and define all rules and factors that aid in categorizing phishing websites, assessed based on performance, accuracy, rule generation volume, and processing speed.

The proposed system effectively implements classification and association algorithms, outperforming the current system in terms of accuracy by reducing the error rate by 30% when incorporating the WHOIS protocol. While no system can detect every phishing website, combining these techniques offers a potential approach to identifying phishing websites effectively. [18] introduced an additional standard-based method for detecting phishing attacks during online account management. Their research demonstrates that the proposed model can accurately discern phishing pages in web-based financial transactions, boasting a precision rate of 99.14% for true positives and only 0.86% for false negatives.

3. 3. Methodology

Python software was used to analyze the web phishing attacks and it involves collecting and analyzing data related to phishing websites or emails [19]. Below is a simplified method for conducting web phishing analysis using Python:

3.1 Data Collection:

3.1.1 Phishing Data Sources: Obtain a dataset of suspected phishing websites or emails. There are various publicly available datasets and APIs for this purpose, or you can create your dataset by collecting phishing samples.

3.1.2 Data Scraping: If you collect data from websites, you may need to use web scraping libraries like Beautiful Soup or Scrapy to extract relevant information, such as website content, URLs, or email text.

3.2 Data Preprocessing:

3.2.1 Data Cleaning: Cleanse and preprocess the collected data. This idea may rely on removing duplicates, handling missing values, and formatting data for analysis.

3.2.2. Feature Extraction: Extract relevant features from the data. For web phishing analysis, features could include the status of URLs, web traffic, links in tags, IP addresses, safe anchors, and textual content from emails or websites.

3.3 Data Analysis:

3.3.1 Exploratory Data Analysis (EDA): Perform EDA to gain insights into the characteristics of phishing data. Visualize data distributions, identify patterns, and detect outliers using libraries like Matplotlib and Seaborn.

3.3.2. Feature Engineering: Create new features or transform existing ones to improve the performance of machine learning models.

3.4 Machine Learning Models:

3.4.1 Classification: Build machine learning models to classify data into phishing and non-phishing categories. Popular classification algorithms for this task include Random Forest, Logistic Regression, Support Vector Machines, and Neural Networks.

3.4.2 Evaluation: Evaluate model performance using accuracy, precision, recall, F1-score, and ROC curves. Utilize cross-validation to ensure model robustness.

3.5 Detection Rules and Heuristics: Develop detection rules or heuristics based on known characteristics of phishing websites or emails. These rules can complement machine learning models.

3.6 Reporting: Create reports or visualizations summarizing the results of your analysis. This event can help stakeholders understand the extent of phishing threats.

3.7 Results Interpretations: Use the insights from the data analysis to make informed decisions, develop strategies, or take actions that address the first objectives.

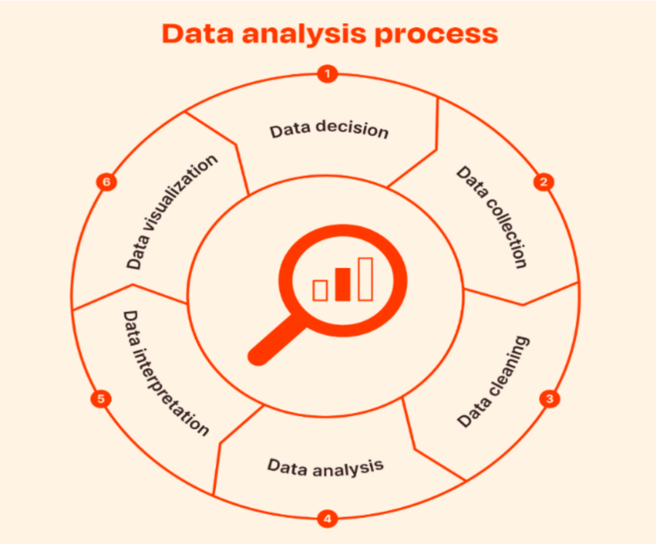


Fig 1. Displays the process taken during the complete analysis.

4. 4. Algorithmic Approaches For Web Phishing Detection

When conducting data analysis for a web phishing project, various calculations and formulas can help extract insights and detect phishing patterns [20] [21]. Here are some standard calculations and formulas relevant to web phishing data analysis:

i. **Descriptive Statistics:**

Average Mean ($\frac{\sum x}{N}$): Calculated as the sum of all values divided by the number of values.

Median:
- if N is odd,
  Median = $X\frac{(N+1)}{2}$
- If N is even, Median = $\frac{1}{2}\left(\left(x\left(\frac{(N)}{2}\right)\right)x\left(\frac{(N)}{2}+1\right)\right)$

Mode: The most frequently occurring value in a dataset.

Variance $\frac{\sum x - mean}{N}$

Standard deviation $\sqrt{Variance}$ Measures the spread or dispersion of data points around the mean.

Range Max−Min, Where Max is the maximum value in the dataset, and Min is the minimum value.

ii. Correlation Analysis:

Pearson Correlation Coefficient: Measures the linear relationship between two continuous variables, ranging from -1 (perfect negative correlation) to 1 (perfect positive correlation). The most used correlation coefficient is the Pearson Correlation Coefficient, denoted as "r." The formula for calculating the Pearson Correlation Coefficient between two variables, X and Y, is as follows:

$$r = \frac{\sum_{i=1}^{n}(Xi - \bar{X})(Yi - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(Xi - \bar{X})^2}\ \sqrt{\sum_{i=1}^{n}(Yi - \bar{Y})^2}}$$

Where:

Xi and Yi : are individual data points of variables X and Y, respectively.

$(\bar{X})$ *and* $(\bar{Y})$: are the means (average) of variables X and Y, respectively.

n: is the number of data points (observations).

Step-by-step explanation of the formula:

5. For each data point (Xi, Yi), calculate the difference between $Xi\ and\ \bar{X}$ and the difference between $Yi\ and\ \bar{Y}$.
6. Multiply these differences for each data point $(Xi - \bar{X}) \times (Yi - \bar{Y})$.
7. Sum up all these products for all data points.
8. Calculate the square root of the sum of the squared differences between $(Xi\ and\ \bar{X})$ and the square root of the sum of the squared differences between $(Yi\ and\ \bar{Y})$.
9. Divide the result from Step 3 by the product of the results from Step 5.

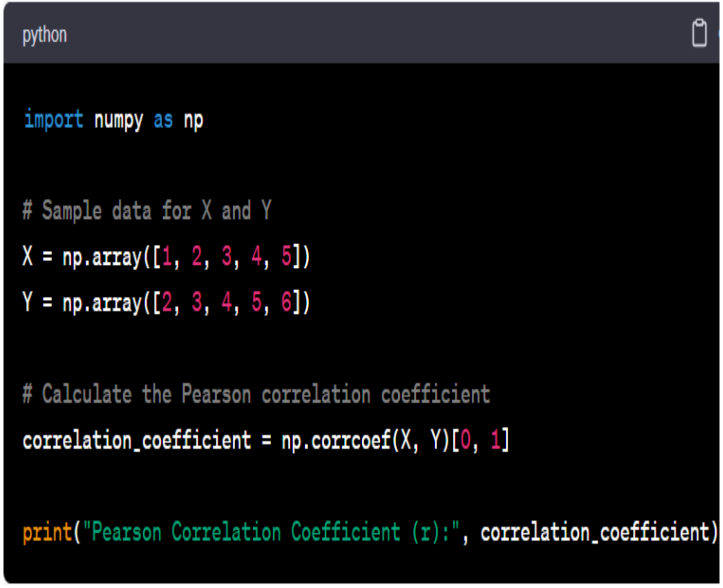
The Pearson Correlation Coefficient "r" will be a value between -1 and 1.

(a) r = 1 indicates a perfect positive linear relationship (as X increases, Y increases).

(b) r = -1 indicates a perfect negative linear relationship (as X increases, Y decreases).

(c) r ≈ 0 indicates little to no linear relationship between X and Y.

You can use libraries like NumPy or SciPy to calculate the Pearson correlation coefficient in Python. Figure 2 shows an example of using NumPy for correlation coefficient calculation.



```python
import numpy as np

# Sample data for X and Y
X = np.array([1, 2, 3, 4, 5])
Y = np.array([2, 3, 4, 5, 6])

# Calculate the Pearson correlation coefficient
correlation_coefficient = np.corrcoef(X, Y)[0, 1]

print("Pearson Correlation Coefficient (r):", correlation_coefficient)
```

Fig 2. Displays programming algorithms using NumPy. function

iii. Entropy (H) = -Σ (p_i * log2(p_i))

Shannon Entropy: Measures the randomness or uncertainty in a dataset. It can assess the diversity of characters or words in URLs or email content.

Higher entropy may indicate suspicious or random patterns.

iv. Phishing Detection Metrics:

True Positive (TP):	The number of phishing instances correctly identified as phishing.
True Negative (TN):	The number of legitimate instances correctly identified as fair.
False Positive (FP):	Legitimate instances incorrectly identified as phishing (Type I error).
False Negative (FN):	Phishing instances incorrectly identified as legitimate (Type II error).
Precision (Positive Predictive Value):	$\frac{\text{TP}}{(TP+FP)}$
Recall (Sensitivity, True Positive Rate):	$\frac{\text{TP}}{(TP+FN)}$
F1-Score:	$\frac{2 * (\text{Precision} * \text{Recall}}{(Precision+Recall)}$
Accuracy	$\frac{(\text{TP}+\text{TN})}{(TP+TN+FP+FN)}$
ROC-AUC: (Receiver Operating Characteristic - Area Under the Curve)	Measures the ability of a model to distinguish between phishing and legitimate instances.
Threshold Optimization:	You may need to adjust the classification threshold of your model to achieve a balance between precision and recall, depending on your project's goals and the cost of false positives and negatives.

v. Time Series Analysis:

For tracking and analyzing phishing attacks over time, you might calculate metrics like the number of phishing incidents per day, week, or month using ARIMA or SARIMA models.

vi. Machine Learning Algorithms:

$$P(Y=1) = \frac{1}{1+e-(b0+b1X1+b2X2+\cdots+bnXn)}$$

Where:	
P(*Y*=1):	is the probability of being phishing.
*b*0+*b*n:	are coefficients learned during training.
*X*1 to *Xn:*	the input features

Machine learning algorithms used for web phishing detection don't typically have single formulas like traditional mathematical equations. Instead, they rely on complex mathematical functions and techniques to learn patterns and make predictions based on training data. Below, is a high-level overview of how some standard machine learning algorithms work in the context of web phishing detection.

vii. Simulations:

If conducting experiments or simulations related to phishing attacks, you may need to calculate probabilities and expected values or simulate various scenarios.

viii. Statistical Tests:

If comparing two or more groups, use statistical tests like t-tests, chi-squared tests, or ANOVA to determine if there are significant differences between groups. The specific calculations and formulas used depend on your dataset, research questions, and the analysis techniques employed. Additionally, Python-libraries like NumPy, Pandas, Scikit-Learn, and SciPy facilitated these calculations and analyses in the web phishing project.

A. Data Preprocessing

Using Python for Data preprocessing is a critical step in the data analysis pipeline. It involves cleaning, transforming, and organizing raw data into a format that is suitable for analysis. Proper data preprocessing helps ensure the accuracy and reliability of your analysis results. Here are the key steps and techniques involved in data preprocessing. The main objective is to use Python in Jupyter Note to take the following steps. Using Pandas library for data manipulation

i. **Data Inspection**:

Examine the data to get a preliminary understanding of its structure and quality. Check for missing values, duplicates, and outliers.

ii. **Data Cleaning and Handling Missing Data**:

Decide how to deal with missing data, which can include: Removing rows or columns with too many missing values. Inputting missing values with a statistical measure (e.g., mean, median, mode) or using more advanced techniques like regression or k-nearest neighbors (KNN) imputation.

iii. **Handling Duplicates**:
Identify and remove duplicate records, which can distort analysis results.

iv. **Data Transformation and Processing**:
Encode categorical variables into a numeric format using techniques like one-hot encoding or label encoding. Normalize or standardize numerical features to bring them to a standard scale, especially when using distance-based algorithms. Apply mathematical transformations to features if they are skewed, such as logarithmic or Box-Cox transformations. Data preprocessing is often an iterative process, and the specific steps you take can vary depending on your dataset and the goals of your analysis. Practical data preprocessing sets the foundation for accurate and meaningful data analysis and modeling.

B. Data Analysis: Interpretation And Visualization

Under data analysis, we are interpreting and visualizing our cleaned data using matplot library, NumPy library, and Seaborn from jupyter note under Python.

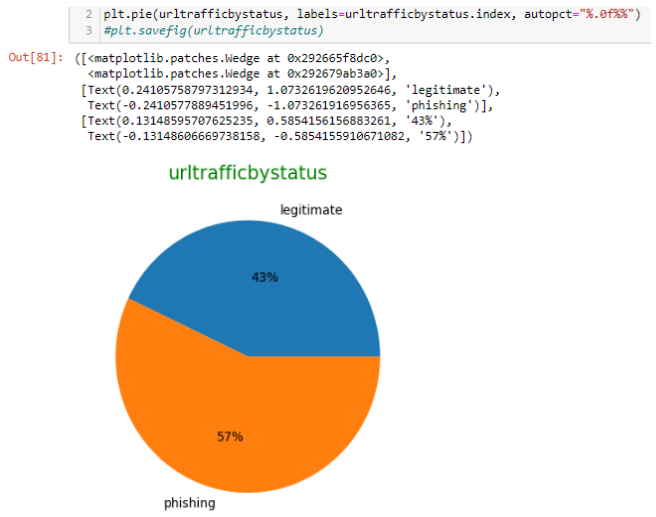
i. **Interpretation:**
We grouped the most important categories, and for this specific project, we used bivariate analysis to interpret and make visuals.

Group by:

a. Data frame by ‘Web Traffic’ and ‘Status' in order to determine if url traffic generated was legitimate or phishing.
b. Dataframe by 'URL' and 'status' in other numbers of URLs that were legitimate and the number that was phishing.

ii. **Visualization:** Visualizing by group

Web Traffic By Status
The pie chart in figure 3 displays the percentage of web traffic across legitimate and phishing uniform resources locator (URL).

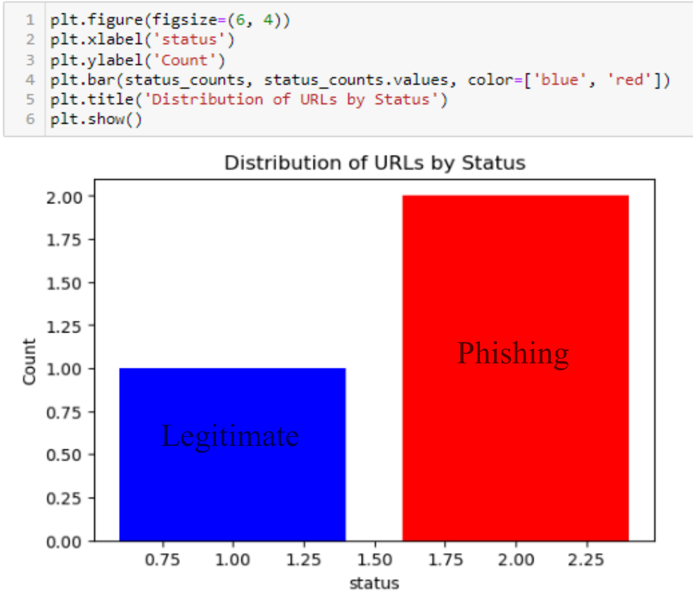


```
plt.pie(urltrafficbystatus, labels=urltrafficbystatus.index, autopct="%.0f%%")
#plt.savefig(urltrafficbystatus)
```

```
Out[81]: ([<matplotlib.patches.Wedge at 0x292665f8dc0>,
           <matplotlib.patches.Wedge at 0x292679ab3a0>],
          [Text(0.24105758797312934, 1.0732619620952646, 'legitimate'),
           Text(-0.2410577889451996, -1.073261916956365, 'phishing')],
          [Text(0.13148595707625235, 0.5854156156883261, '43%'),
           Text(-0.1314860666973815８, -0.585415591067108２, '57%')])
```

Fig 3. Displays how much traffic the phishing sites get compared to the legitimate sites.

URL BY STATUS



```
plt.figure(figsize=(6, 4))
plt.xlabel('status')
plt.ylabel('Count')
plt.bar(status_counts, status_counts.values, color=['blue', 'red'])
plt.title('Distribution of URLs by Status')
plt.show()
```

Fig 4. Displays the number of phishing URLs compared to legitimate URLs.

10. 5. Results

Fig. 3 demonstrates the accuracy of the enormous amount of traffic being generated by the phishing websites and has been reanalyzed by machine learning algorithms from Scikit Learn and mathematical equations in NumPy to verify the accuracy of the data and information being produced. The same thing could be said for Fig. 4, showing the huge number of URLs that aren't legitimate but phishing, as I will be displaying the full results of our analysis for including the safety anchor and links in tags; stay informed about the latest phishing techniques, and foster a security-conscious culture in your organization to ensure a robust defense against phishing attacks.

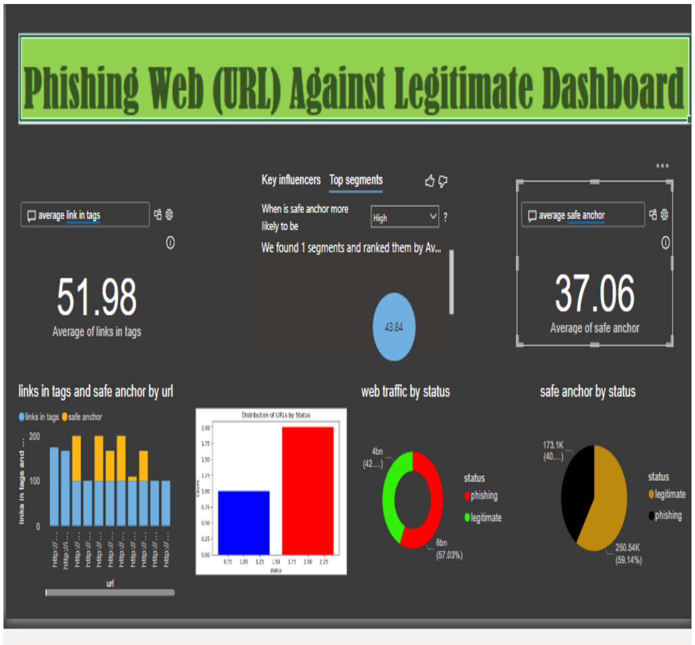


Fig 5. Dashboard for phishing and legitimate URLs

Suppose your web phishing dashboard is showing legitimate URLs with higher safety scores than phishing URLs. In that case, it shows that your phishing detection and prevention measures are effective in identifying and classifying legitimate websites accurately. This event is a positive outcome, as it suggests that the system is successfully distinguishing between safe and potentially malicious URLs.

Here's how you can interpret and use these results effectively:

A. Validation of Safety Measures:

The higher safety scores for legitimate URLs validate the effectiveness of your safety measures, such as email filtering, web filtering, or user education programs.

B. Confidence in Legitimate URLs:

Users and administrators can have greater confidence in URLs classified as legitimate. This fact can reduce the chances of false positives, where legitimate websites are incorrectly flagged as phishing sites.

C. Focus on Phishing URLs:

While the focus is on legitimate URLs with high safety scores, it's equally important to monitor and investigate phishing URLs with lower safety scores. These may represent emerging threats or new phishing tactics that need attention.

D. Continuous Monitoring:

Keep monitoring the dashboard regularly to identify any changes in phishing trends or the safety scores of legitimate websites. Rapidly evolving threats may require adjustments to your security measures.

E. User Education:

Use the dashboard to educate users about the importance of verifying the legitimacy of websites, especially when they have URLs with lower safety scores. Encourage reporting of suspicious URLs.

F. Feedback and Improvements:

Collect feedback from users and security analysts regarding the accuracy of the safety scores. This feedback can help fine-tune your detection algorithms and improve the overall effectiveness of your phishing prevention system.

G. Response to Phishing URLs:

For phishing URLs with lower safety scores, ensure that you have an incident response plan in place. Promptly investigate and mitigate any confirmed phishing attempts to minimize potential risks.

H. Benchmarking and Goal Setting:

Use the dashboard data to set benchmarks and goals for your security team. Aim to continuously improve the accuracy of phishing detection and reduce false negatives.

11. 6. Summary

Distinguishing between legitimate and phishing URLs is critical in today's digital landscape, where cyber threats continue to evolve. Understanding the key indicators and employing best practices can help individuals and organizations protect themselves against phishing attacks. In conclusion, the ability to differentiate between legitimate and phishing URLs is an essential skill in today's digital age. Phishing attacks continue to pose significant threats, making it imperative for individuals and organizations to remain vigilant. By following best practices, inspecting URLs, verifying content, and leveraging available tools, the risk can be reduced to avoid falling victim to phishing schemes.

References

1. O. Adekunle *et al.*, “A Review of Cybersecurity as an Effective Tool for Fighting Identity Theft across United States,” *Int. J. Cybern. Inform.*, vol. 12, no. 5, pp. 31–42, Aug. 2023, doi: 10.5121/ijci.2023.120504.
2. P. F. Likarish, Early detection of malicious web content with applied machine learning. The University of Iowa, 2011.
3. E. O. Paul *et al.*, “Cybersecurity Strategies for Safeguarding Customer’s Data and Preventing Financial Fraud in the United States Financial Sectors,” *Int. J. Soft Comput.*, vol. 14, no. 3, pp. 01–16, Aug. 2023, doi: 10.5121/ijsc.2023.14301.
4. B. G. Bokolo, L. Chen, and Q. Liu, “Deep Learning Assisted Cyber Criminal Profiling,” in *2023 IEEE 6th International Conference on Big Data and Artificial Intelligence (BDAI)*, Jiaxing, China: IEEE, Jul. 2023, pp. 226–231. doi: 10.1109/BDAI59165.2023.10257003.
5. [5] A. Adefabi, S. Olisah, C. Obunadike, O. Oyetubo, E. Taiwo, and E. Tella, “Predicting Accident Severity: An Analysis of Factors Affecting Accident Severity Using Random Forest Model,” *Int. J. Cybern. Inform.*, vol. 12, no. 6, pp. 107–121, Oct. 2023, doi: 10.5121/ijci.2023.120609.
6. C. Obunadike, A. Adefabi, S. Olisah, D. Abimbola, and K. Oloyede, “Application of Regularized Logistic Regression and Artificial Neural Network Model for Ozone Classification across El Paso County, Texas, United States,” *J. Data Anal. Inf. Process.*, vol. 11, no. 03, pp. 217–239, 2023, doi: 10.4236/jdaip.2023.113012.
7. Z. Dong, A. Kapadia, J. Blythe, and L. J. Camp, “Beyond the lock icon: real-time detection of phishing websites using public key certificates,” presented at the 2015 APWG Symposium on Electronic Crime Research (eCrime), IEEE, 2015, pp. 1–12.
8. B. G. Bokolo, L. Chen, and Q. Liu, “Detection of Web-Attack using DistilBERT, RNN, and LSTM,” in *2023 11th International Symposium on Digital Forensics and Security (ISDFS)*, Chattanooga, TN, USA: IEEE, May 2023, pp. 1–6. doi: 10.1109/ISDFS58141.2023.10131822.
9. Z. Alkhalil, C. Hewage, L. Nawaf, and I. Khan, “Phishing attacks: A recent comprehensive study and a new anatomy,” *Front. Comput. Sci.*, vol. 3, p. 563060, 2021.
10. C. Ratcliffe, B. G. Bokolo, D. Oladimeji, and B. Zhou, “Detection of Anti-forensics and Malware Applications in Volatile Memory Acquisition,” in *Advances and Trends in Artificial Intelligence. Theory and Practices in Artificial Intelligence*, vol. 13343, H. Fujita, P. Fournier-Viger, M. Ali, and Y. Wang, Eds., in Lecture Notes in Computer Science, vol. 13343. , Cham: Springer International Publishing, 2022, pp. 516–527. doi: 10.1007/978-3-031-08530-7_44.
11. E. Ulqinaku, D. Lain, and S. Capkun, “2FA-PP: 2nd factor phishing prevention,” presented at the Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, 2019, pp. 60–70.
12. B. B. Gupta, A. Tewari, A. K. Jain, and D. P. Agrawal, “Fighting against phishing attacks: state of the art and future

challenges," *Neural Comput. Appl.*, vol. 28, pp. 3629–3654, 2017.

13. B. G. Bokolo and Q. Liu, "Cyberbullying Detection on Social Media Using Machine Learning," in *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Hoboken, NJ, USA: IEEE, May 2023, pp. 1–6. doi: 10.1109/INFOCOMWKSHPS57453.2023.10226114.
14. [14] P. Parekh, K. Parmar, and P. Awate, "Spam URL detection and image spam filtering using machine learning," *Comput Eng*, 2018.
15. [15] M. Khonji, Y. Iraqi, and A. Jones, "Phishing Detection: A Literature Survey," *IEEE Commun. Surv. Tutor.*, vol. 15, no. 4, pp. 2091–2121, 2013, doi: 10.1109/SURV.2013.032213.00009.
16. [16] N. Abdelhamid, A. Ayesh, and F. Thabtah, "Phishing detection based Associative Classification data mining," *Expert Syst. Appl.*, vol. 41, no. 13, pp. 5948–5959, Oct. 2014, doi: 10.1016/j.eswa.2014.03.019.
17. [17] H. Sampat, M. Saharkar, A. Pandey, and H. Lopes, "Detection of phishing website using machine learning," *Int Res J Eng TechnolIRJET*, vol. 5, no. 3, 2018.
18. [18] M. Moghimi and A. Y. Varjani, "New rule-based phishing detection method," *Expert Syst. Appl.*, vol. 53, pp. 231–242, 2016.
19. [19] B. G. Bokolo, P. Onyehanere, E. Ogegbene-Ise, I. Olufemi, and J. N. A. Tettey, "Leveraging Machine Learning for Crime Intent Detection in Social Media Posts," in *AI-generated Content*, vol. 1946, F. Zhao and D. Miao, Eds., in Communications in Computer and Information Science, vol. 1946. , Singapore: Springer Nature Singapore, 2024, pp. 224–236. doi: 10.1007/978-981-99-7587-7_19.
20. N. J. Gogtay and U. M. Thatte, "Principles of correlation analysis," *J. Assoc. Physicians India*, vol. 65, no. 3, pp. 78–81, 2017.
21. D. George and P. Mallery, "Descriptive statistics," in *IBM SPSS Statistics 25 Step by Step*, Routledge, 2018, pp. 126–134.