International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0505
e ISSN: 2584-2854
Volume: 03 Issue: 11
November 2025
Page No: 3185 - 3190

Detecting Malware Website Using Machine Learning Methods and Techniques

Dr. D. Gayathri[1]
[1]*Assistant Professor, Department of Computer Science, Government Arts & Science College, Mettur, Salem, TamilNadu, 636401, India.*
Emails: *dgayathrigascm@gmail.com*[1]

Abstract
A malware website is a site designed to harm users by installing malicious software (malware) on their devices, stealing data, or redirecting them to other harmful sites. A malware website spreads malware, infects the victim's system, and steals important information to harm the user. The global lockdown will see an increase in and shift toward using internet services while staying at home in 2020. As a result, businesses suffered significant data breaches and the number of cybercrimes committed by criminals increased. Malware URLs and threat types must be located in order to stop these attacks. The majority of malware web pages can be identified by static properties that describe these behaviors because they import exploits from distant resources and conceal exploit code. In recent years, a number of models and approaches have been proposed to identify such phishing URLs. In this paper, a machine learning strategy based on a machine learning model for the most accurate detection of malware websites is reviewed and proposed. In addition, we carry out a reconnaissance on the URL to supply additional details regarding the port status, directories, and subdomains of the website.

Keywords: *URL, most accurate detection and a machine learning strategy.*

1. Introduction
Malware website detection methods include signature-based detection, behavioral analysis, machine learning, and sandboxing. Signature-based detection compares files to a known database of malware, while behavioral and heuristic analysis looks for suspicious code patterns or actions. Machine learning and AI can analyze vast amounts of data to identify patterns of unknown threats, and sandboxing safely tests suspicious software in an isolated virtual environment.

Traditional methods

- **Signature-based detection:** This traditional method works by comparing files against a database of known malware "signatures" (unique identifiers). It is effective against known threats but struggles with new or polymorphic malware.
- **Heuristic analysis:** This method uses algorithms to identify suspicious code patterns, even if the specific malware isn't in the database. It looks for general characteristics of malware to detect novel threats.

Advanced methods

- **Behavioral analysis:** This technique monitors the actions of a program or website in real-time, looking for anomalous or malicious behavior. An Intrusion Detection System (IDS) can be used to detect suspicious patterns in network traffic.
- **Sandboxing:** Suspicious files are run in a safe, isolated virtual environment called a sandbox to observe their behavior without risking the actual system. This allows for dynamic analysis of a program's capabilities.
- **Machine Learning (ML) and AI:** These techniques analyze large datasets to identify patterns, anomalies, and trends associated

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0505
e ISSN: 2584-2854
Volume: 03 Issue: 11
November 2025
Page No: 3185 - 3190

with malicious activity, which helps in detecting previously unknown threats. Algorithms can classify URLs or code as safe or malicious based on their features.

Other supporting techniques

- **Static analysis:** This involves examining the code or structure of a file without executing it to look for malicious characteristics.
- **Intrusion Detection/Prevention Systems (IDS/IPS):** These systems monitor network traffic for suspicious patterns, and IPS can also actively block malicious activity.
- **DNS analysis:** Analyzing DNS query logs can help identify malicious activities by spotting unusual patterns.
- **Reputation-based detection:** This method assesses a website's or file's reputation based on past behavior and known associations with malicious activity.

The rapid growth of the internet has led to an increase in cyber threats, with malware websites posing significant risks to users. Malicious websites can infect devices, steal sensitive information, or distribute harmful software, making it essential to detect and block these sites quickly. Traditional methods of detecting such websites, such as manual analysis or signature-based approaches, are often ineffective due to the constant evolution of cyberattacks and the sheer volume of websites to monitor. Machine learning (ML) has emerged as a powerful tool for automating the detection of malware websites. By training algorithms on large datasets of both benign and malicious websites, ML models can learn to identify patterns and features that differentiate harmful sites from legitimate ones. These models can analyze various website characteristics, such as URL structure, domain information, content behavior, and traffic patterns, to classify sites as either safe or suspicious. This approach has several advantages, including scalability, adaptability to new threats, and the ability to analyze large datasets in real-time. By leveraging machine learning techniques, it is possible to develop an automated system capable of detecting malware websites with high accuracy and efficiency, offering a robust solution to enhance online security. This paper explores the implementation of such a system, detailing the data collection, feature extraction, model training, and evaluation processes, as well as the challenges and opportunities in using machine learning for cybersecurity

1.1. Malicious URL Detection

The improvement of unused communication advances has had a noteworthy affect on trade development and advancement in an assortment of settings. The World Wide Web has consistently expanded in significance. Shockingly, modern, modern strategies of assaulting and duping clients come with mechanical progressions. These assaults can incorporate noxious websites that offer fake merchandise and uncover delicate data, coming about in the robbery of cash and personality, as well as the establishment of malware on the user's framework. Assaults can be carried out in a wide extend of ways, counting unequivocal hacking endeavors, drive-by abuses, phishing, watering gaps, social designing, man-in-the-middle assaults, SQL infusions, misfortune or burglary administrations, and so on. There are numerous distinctive sorts of assaults, and unused sorts of assaults are made each day. It is troublesome to arrange a incredible system to recognize the security breaks in the advanced world. The restrictions of the boycotting strategy are getting more regrettable and more awful at an exponential rate, fair like the unused security dangers. The larger part of these assaults are well-known due to the dispersal of compromised URLs. Compromised URLs are utilized for computerized ambushes which are known as pernicious URLs. One third of all websites are noxious in nature, agreeing to measurements. The convention identifier, which indicates the convention to utilize, and the asset title, which indicates the IP address or space title where the asset is found, are the two essential components of a URL. A colon and two forward cuts isolated the asset title from the convention identifier. The detection of malicious URLs is a critical component of cybersecurity, as these URLs are often used to deliver phishing attacks, malware, ransomware, or other harmful activities. Malicious URLs are crafted to deceive users into visiting harmful websites that steal personal data, distribute viruses, or exploit

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0505
e ISSN: 2584-2854
Volume: 03 Issue: 11
November 2025
Page No: 3185 - 3190

vulnerabilities in web applications. As the number of these attacks continues to rise, effective detection methods are necessary to protect users and organizations from these threats. Malicious URL detection typically involves analyzing the URL's structure, content, and behavior to identify suspicious or harmful characteristics. The primary challenge lies in the constantly evolving tactics used by cybercriminals to disguise their malicious URLs and avoid detection. Traditional methods, such as blacklist-based detection or heuristic analysis, often fail to keep up with new threats, especially as attackers continuously change their strategies [1].

1.2. Definition of the Problem

We model the problem of malicious URL detection as a two-class prediction binary classification task: malicious as opposed to benign. In particular, given a data set containing T URLs, f(u1; y1); : : : ; (uT ; yT)g, where ut for t equals 1; T addresses a URL from the preparation information, and yt 2 f1; The label that corresponds to this is 1g, where yt = 1 denotes a malicious URL and yt = 1 denotes a benign URL. The two most important aspects of automated malicious URL detection are: 1) Representation of Features: Getting the right representation of the features: ut ! xt, where xt 2 Rd is the URL's d-dimensional feature vector; what's more, 2) AI: Learning an expectation capability f : Rd ! R, which makes accurate feature presentations to predict the class assignment for any URL instance x.

2. System Study

In order to train an ML model, training data must be provided to the ML algorithm—also known as the learning algorithm. The model artifact produced by the training process is referred to as an ML model. A target or target attribute—the correct response—must be present in the training data. An ML model that captures these patterns is produced by the learning algorithm after it finds patterns in the training data that map the attributes of the input data to the target (the answer you want to predict). The ML model can be used to predict new data for which the target is unknown. Let's say you want to train an ML model to determine whether an email is spam or not. Amazo ML would be provided with training data that contained emails for which you knew the target (a label that indicated whether or not an email was spam). The system implementation for detecting malware websites using machine learning involves several key stages. First, data collection is crucial, where datasets containing both malicious and benign websites are gathered. This data can be sourced from public malware databases, web crawlers, or DNS and web traffic logs. Once the data is collected, feature extraction begins, which involves identifying and extracting relevant features such as URL length, domain registration details, SSL certificate information, and website content characteristics. These features help distinguish between legitimate websites and those potentially hosting malware. Following feature extraction, data preprocessing is necessary to clean the data, handle missing values, normalize features, and encode categorical data into a suitable format for machine learning models Next, model selection takes place, where different machine learning algorithms are considered, such as decision trees, support vector machines (SVM), logistic regression, or neural networks. These models are chosen based on the nature of the data and the complexity of the problem. After selecting the model, model training occurs using historical data, typically splitting the dataset into training and testing sets. Cross-validation techniques and hyper parameter tuning are used to optimize the model's performance. Once the model is trained, it is evaluated using metrics like accuracy, precision, recall, F1-score, and the ROC curve to assess its effectiveness in distinguishing malicious websites from legitimate ones. Once the model is successfully trained and evaluated, it can be deployed into a real-time detection system. The deployment involves integrating the trained model into a web filtering or security system, which continuously scans websites and flags malicious ones in real time. Continuous monitoring and updating are also essential to keep the system up-to-date with emerging threats and to improve accuracy. Retraining the model with new data, performance tracking, and user feedback integration ensure the system remains effective in identifying evolving malware websites. By automating this process, machine learning systems offer a robust approach to protecting users from

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0505
e ISSN: 2584-2854
Volume: 03 Issue: 11
November 2025
Page No: 3185 - 3190

online security threats [2].

3. Existing System

They are unable to detect new threats due to their inability to maintain an exhaustive list of all malicious URLs, as new URLs are frequently generated. Many organizations rely on human experts to manually review websites and identify potential threats. This approach, while effective in some cases, is labor-intensive and slow, making it impractical for real-time detection, especially considering the massive number of websites that are created daily. Some existing systems have started integrating machine learning models to detect malware websites. These models often focus on basic features, such as domain registration details, URL structure, or the presence of known malware indicators in the website's content. However, these models still face challenges in terms of accuracy, false positives, and the ability to keep up with rapidly evolving attack strategies. Heuristic methods attempt to identify malicious websites by examining certain characteristics or behaviors typical of harmful sites. These features can include suspicious URL patterns, unusual site content, or patterns of user interaction that are indicative of phishing or malware distribution. While heuristic methods can catch some previously unknown threats, they often produce false positives and may not be comprehensive enough to identify all malicious websites.

3.1. Disadvantages

- URL blacklisting is ineffective for new malicious URLs.
- It takes time to analyze malicious URLs and propagate a blacklist to end users.
- Suffers from nontrivial high false negatives.
- Blacklist features alone do not have as good performance as other features [3].

4. Proposed System

The proposed system for malicious URL detection aims to address the limitations of existing detection methods by leveraging advanced machine learning techniques. Unlike traditional approaches that rely on predefined rules, blacklists, or heuristic analysis, this system uses machine learning models to automatically analyze and classify URLs based on their inherent characteristics. The goal is to develop an efficient and scalable solution that can detect new, unknown malicious URLs with high accuracy and minimal false positives. By extracting good feature representations of URLs and training a prediction model on training data of both malicious and benign URLs, these strategies attempt to analyze the information of a URL and its corresponding websites or webpages. Static and dynamic features are the two types of features that can be used. In static analysis, we analyze a webpage using information that is accessible without having to execute the URL. The underlying assumption is that these features are distributed differently between benign and malicious URLs. A prediction model that can make predictions about new URLs can be built with this distribution information. Static analysis techniques have been extensively investigated by utilizing machine learning techniques due to the ability to generalize to all kinds of threats and the relatively safer environment for extracting important information.

4.1. Advantages of Proposed System

- By extracting good feature representations of URLs, this method attempts to analyze the information of a URL and the websites or webpages that correspond to it.
- Using training data of both malicious and benign URLs to train a prediction model.
- Using this information about the distribution, a prediction model that can make predictions about new URLs can be built [4].

5. System Implementation

5.1. Data Collection

This engineering-oriented phase aims to collect the majority, if not all, pertinent information regarding the URL. This includes information about the host, the content of the website, such as HTML and JavaScript, popularity information, the direct features of the URL, such as the URL String, and whether the URLs are on a blacklist.

5.2. Preprocessing

The unstructured information about the URL, such as a textual description, is formatted correctly and transformed into a numerical vector during this phase so that machine learning algorithms can use it. The BoW, on the other hand, is used to represent textual or lexical content, while the numerical information,

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0505
e ISSN: 2584-2854
Volume: 03 Issue: 11
November 2025
Page No: 3185 - 3190

on the other hand, can be used as is. Additionally, some data normalization, such as Z-score normalization, may frequently be utilized to address the scaling problem.

5.3. Feature Extraction

Analysts have proposed an assortment of highlights that can be utilized to give valuable data for the reason of recognizing pernicious URLs. We classify these characteristics as takes after: Host-based highlights, content-based highlights, boycott highlights, URL-based lexical highlights, and others All have focal points and impediments, and whereas a few are exceptionally educator, securing these highlights can be exceptionally exorbitant. Additionally, pre-processing troubles and security concerns shift depending on the include.

5.4. Labeling

Analysts have proposed a assortment of highlights that can be utilized to give valuable data for the reason of identifying pernicious URLs. We classify these characteristics as takes after: Host-based highlights, content-based highlights, boycott highlights, URL-based lexical highlights, and others All have preferences and drawbacks, and whereas a few are exceptionally teacher, procuring these highlights can be exceptionally expensive. So also, pre-processing troubles and security concerns shift depending on the include [5].

5.5. Validation and Prediction

When taking into account a large number of trees, the Random Forest algorithm prevents overfitting. The main advantage that Random Forest has over other algorithms is that it can handle missing values by itself. The mean of all the decision tree accuracies will determine the final random forest's accuracy Shown in Figure 1.

6. System Architecture

Figure 1 System Architecture

7. Future Work

A rule-based prediction based on a URL's content analysis is something we would like to incorporate. For phishing URL detection, therefore, a comprehensive solution would be provided by combining a rule-based URL content analyzer with a classification-based lexical analyzer [6].

Conclusions

From the set of URLs that contain both benign and phishing URLs, we have investigated how effectively phishing URLs can be classified. The dataset's randomization, feature engineering, feature extraction using lexical analysis of host-based features, and statistical analysis have also been

International Research Journal on Advanced Engineering and Management
https://goldncloudpublications.com
https://doi.org/10.47392/IRJAEM.2025.0505
e ISSN: 2584-2854
Volume: 03 Issue: 11
November 2025
Page No: 3185 - 3190

discussed. In addition, we have conducted the comparative study with a variety of classifiers and discovered that the results are nearly identical across all of them. Additionally, we observed that dataset randomization resulted in a significant improvement in the classifier's accuracy as well as an excellent optimization. Utilizing straightforward regular expressions, we have adopted a straightforward strategy for extracting the features from the URLs. It's possible that the system's accuracy can be further enhanced by experimenting with additional features. Since the URLs list in the dataset used in this paper may be a little out of date, regular continuous training with a new dataset would significantly improve model accuracy and performance. Since it is difficult to train an ML classifier based on its content-based features, we did not use the content-based features in our experiment because the main issue with the content-based strategy for detecting phishing URLs is the lack of availability of phishing websites and their short lifespan.

References

[1]. Daron Acemoglu, Munther A Dahleh, Ilan Lobel, and Asuman Ozdaglar (2021), Bayesian learning in social networks. The Review of Economic Studies, 78(4):1201–1236.

[2]. Daron Acemoglu, Asuman Ozdaglar, and Ali ParandehGheibi (2021), Spread of (mis) information in social networks. Games and Economic Behavior, 70(2):194–227.

[3]. Ross Anderson and Tyler Moore (2006). The economics of information security. Science, 314(5799):610–613.

[4]. Fabr´ıcio Benevenuto, Tiago Rodrigues, Meeyoung Cha, and Virg´ılio Almeida. Characterizing user behavior in online social networks. In Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference, pages 49–62. ACM, 2021.

[5]. Andre Bergholz, Jeong Ho Chang, Gerhard Paaß, Frank Reichartz, and Siehyun Strobel (2008), Improved phishing detection using model-based features. In CEAS.

[6]. Leyla Bilge, Engin Kirda, Christopher Kruegel, and Marco Balduzzi (2011), Exposure: Finding malicious domains using passive dns analysis. In NDSS.

Authors Profile

Dr. D. Gayathri has completed her M.C.A form J.K.K Nataraja College of Arts and Science, Komarapalayam, affiliated with Periyar University. She completed her M. Phil in Periyar University, Salem in the year 2005. She was qualified in SET in 2016. She received her Ph.D., in Periyar University, Salem in the year 2020. She has published around 20 papers in reputed journals and National and International Conferences. Now she is working as Assistant Professor, Department of Computer Science in Government Arts and Science College, Mettur, Salem Dt. Her area of interest includes Information Security, Machine Learning and Artificial Intelligence.