Due to the COVID-19 pandemic, the phishing attacks have increased exponentially. This has put a large number of people at risk. Earlier, a rule-based approach was used to detect these malicious URLs in which companies maintained a blacklist of malicious URLs. For every request, a query was made to the blacklist database to get the status of that particular URL. However, due to an increase in URLs and sites this method became ineffective and obsolete. More people are now connecting to a larger number of sites from around the world. Thus, we need a more intelligent and scalable solution. Machine learning based approaches for malicious URL detection have been around for quite some time now and recently they have picked up momentum due to a rapid increase in the number of websites and also due to an increase in the APIs providing users new ways of interacting with the sites.

URL is the abbreviation of Uniform Resource Locator, which is the global address of documents and other resources on the World Wide Web. A URL has two main components: (i) protocol identifier (indicates what protocol to use) (ii) resource name (specifies the IP address or the domain name where the resource is located). The protocol identifier and the resource name are separated by a colon and two forward slashes. The protocol identifier consists of HTTP or HTTPS. Https means a secured and encrypted SSL certificate-based communication that prevents MITM (man in the middle).

Many assume that an HTTPS connection means that the site is secure. In fact, HTTPS is increasingly being used by malicious sites, especially phishing ones.

Compromised URLs that are used for cyber-attacks are termed as malicious URLs. In fact, it was noted that close to one-third of all websites are potentially malicious in nature, demonstrating rampant use of malicious URLs to perpetrate cyber-crimes. A Malicious URL or a malicious web site hosts a variety of unsolicited content that can potentially damage the client devices. The various types of attacks are drive-by download or DGA.

Below we describe and analyze one such ML-based approach:

**1.)Data Acquiring and Analysis :**

The data set generally consists of large number of URLs which have labeled as benign or malicious. However, we can create a more informative data set by adding more feature like :

a) HOST based features:

These include features like hostname, WHOIS, Domain IP etc. Host-based features are obtained from the host-name properties of the URL. They allow us to know the location, identity, and the management style and properties of malicious hosts.

b) Lexical features are concerned with the physical appearance of a URL. This includes the length of URL, special characters present in URL, length of the hostname, and Top-level domain(TLD). A more realistic data can be formed if we combine all the above features along with know sources like URLs with high Alexa ranks which are legitimate and Phish tank URLs that are known to be malicious.

**2.)Preprocessing:**

Often in real-world the data set is imbalanced towards more benign samples. We can use under-sampling or some other techniques but in real life scenario these models will face more benign than malicious URLs So, this strategy might help in reducing false positives.

We should then tokenize the given URLs based on certain parameters like "/-" .We can remove frequently used terms that dont affect the prediction like "com" and "www" from the given URLs.

Machine learning algorithms require numeric inputs So, we pass this tokenized list to the count vectorizer or TF-IDF vectorizer to convert text words into numeric feature vectors.

The vectorizer objects provided by Scikit-Learn are quite reliable right out of the box, they allow us to perform all the above steps at once efficiently, and even apply preprocessing and rules regarding the number and frequency of tokens using NLP techniques like tf-idf.We should go for **tf-idf** scores instead of using bag of words classification since there are words in URLs that are more important than other words e.g presence of PE file names like exe , .so or dylib etc.

**3.)Training and Evaluating:**

Since machine learning is an iterative process we should never rely on a single algorithm. Logistic regression and naïve Bayes has been proven to be effective in these type of problems. The F1 is an effective evaluation metric. It is the harmonic mean of precision and recall.Therefore, this score takes both false positives and false negatives into account. We can also compute accuracy for the algorithms involved.

**Conclusion:** Malicious URL detection plays a critical role in many cybersecurity applications, and clearly machine-learning approaches are a promising direction. In this article, we conducted a comprehensive and systematic survey on Malicious URL Detection using machine learning techniques. In particular, we offered a systematic formulation of Malicious URL detection from a machine learning perspective, and then detailed the discussions of existing studies for malicious URL detection