Project Report

Phishing:-

Phishing through links is a deceptive tactic used by cybercriminals to trick individuals into divulging sensitive information such as login credentials, financial details, or personal information. It typically involves sending fraudulent emails, messages, or social media posts that contain links to malicious websites designed to mimic legitimate ones. These fraudulent websites often closely resemble authentic sites, such as online banking portals, social media platforms, or email login pages.

Working:-

Creation of a Phishing Link:-
Cybercriminals create a fake website that closely resembles a legitimate one, often copying the design, layout, and branding elements. They then generate a link to this fake website.

Distribution of Phishing Links:-

The perpetrators send out phishing emails, messages, or social media posts containing these deceptive links. They may use various tactics to make the messages appear legitimate, such as using the logo and branding of well-known companies or organizations, creating urgency or fear to prompt immediate action, or impersonating trusted contacts.

Lure and Deception:-

The messages typically contain enticing offers, urgent requests, or alarming warnings to prompt recipients to click on the provided link. For example, a phishing email might claim that there has been unauthorized activity on the recipient's bank account and urge them to click on a link to verify their account details.

Redirect to Fake Website:-

When the recipient clicks on the phishing link, they are redirected to the fraudulent website, which closely resembles the legitimate one they were expecting. This fake website is designed to capture any information entered by the victim, such as usernames, passwords, credit card numbers, or other personal details.

Data Harvesting:-

Once on the fake website, victims may be prompted to enter their credentials or other sensitive information. When they do so, this information is captured by the cybercriminals and can be used for various malicious purposes, including identity theft, financial fraud, or unauthorized access to accounts.

Phishing through links poses several threats:

Identity Theft:-

By tricking individuals into providing their personal information, cybercriminals can steal identities and use the stolen information to open fraudulent accounts, make unauthorized purchases, or commit other forms of financial fraud.

Financial Loss:-

Phishing attacks can result in financial losses for individuals or organizations, especially if sensitive financial information such as credit card numbers or banking credentials is compromised.

Data Breaches:-

Phishing attacks can lead to data breaches if cybercriminals successfully obtain access to sensitive information stored on corporate networks or online accounts.

Reputation Damage:-

Organizations targeted by phishing attacks may suffer reputational damage if customers or clients lose trust in their ability to protect sensitive information.

Malware Infection:-

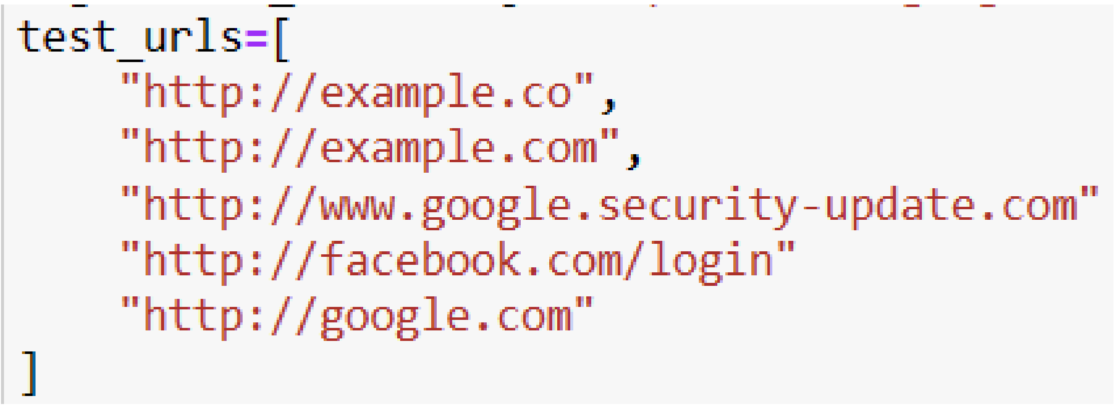
In addition to stealing information directly through phishing websites, cybercriminals may also use phishing links to distribute malware. Clicking on a malicious link could result in the unwitting download and installation of malware onto the victim's device, leading to further compromise of their security and privacy.

Introduction:

In an increasingly digital world, where online communication and transactions have become ubiquitous, the threat of phishing attacks looms large. Phishing, a form of cybercrime wherein malicious actors impersonate legitimate entities to deceive individuals into divulging sensitive information, remains a prevalent and insidious threat to cybersecurity.The proliferation of phishing attacks underscores the need for robust defenses to safeguard against the exploitation of unsuspecting users. Recognizing this imperative, the present project endeavors to develop a sophisticated phishing link scanner tool. This tool represents a proactive measure in the ongoing battle against cyber threats, particularly those aimed at exploiting human vulnerabilities.The overarching goal of this endeavor is to equip users with a reliable mechanism for assessing the legitimacy of URLs encountered in their online interactions. By harnessing a combination of advanced techniques and algorithms, the phishing link scanner aims to empower users to make informed decisions when confronted with suspicious links, thereby fortifying their defenses against potential phishing attacks.In this introductory phase, we outline the objectives, methodology, and expected outcomes of the project. Through meticulous research, algorithm development, and rigorous testing, we endeavor to create a tool that not only accurately identifies phishing threats but also enhances user awareness and resilience in the face of evolving cyber risks.As we embark on this endeavor, we remain steadfast in our commitment to leveraging technology for the betterment of cybersecurity. Through collaboration, innovation, and a relentless pursuit of excellence, we aspire to contribute to a safer and more secure digital landscape for individuals and organizations alike.

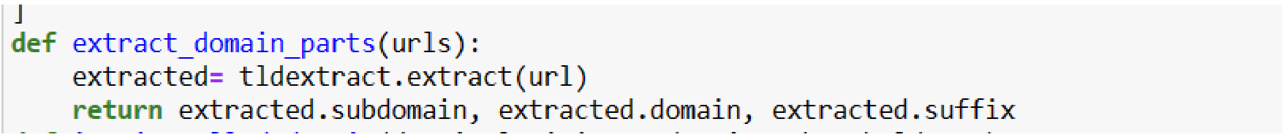
Methodology: The phishing link scanner employs a combination of techniques to assess the legitimacy of URLs. These techniques include:

1. **URL Analysis**: The tool analyzes the structure and components of the URL to identify any suspicious elements, such as unusual domain names, subdomains, or excessive query parameters.



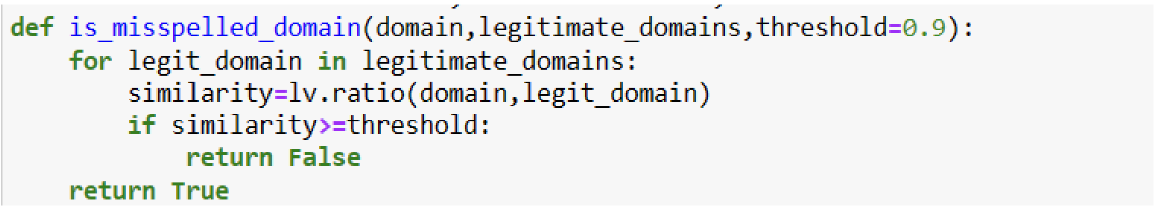
```
test_urls=[
    "http://example.co",
    "http://example.com",
    "http://www.google.security-update.com"
    "http://facebook.com/login"
    "http://google.com"
]
```

2. **Domain Reputation Check**: It queries domain reputation databases and services to assess the trustworthiness of the domain associated with the URL. This step helps identify known phishing domains or those associated with malicious activities.



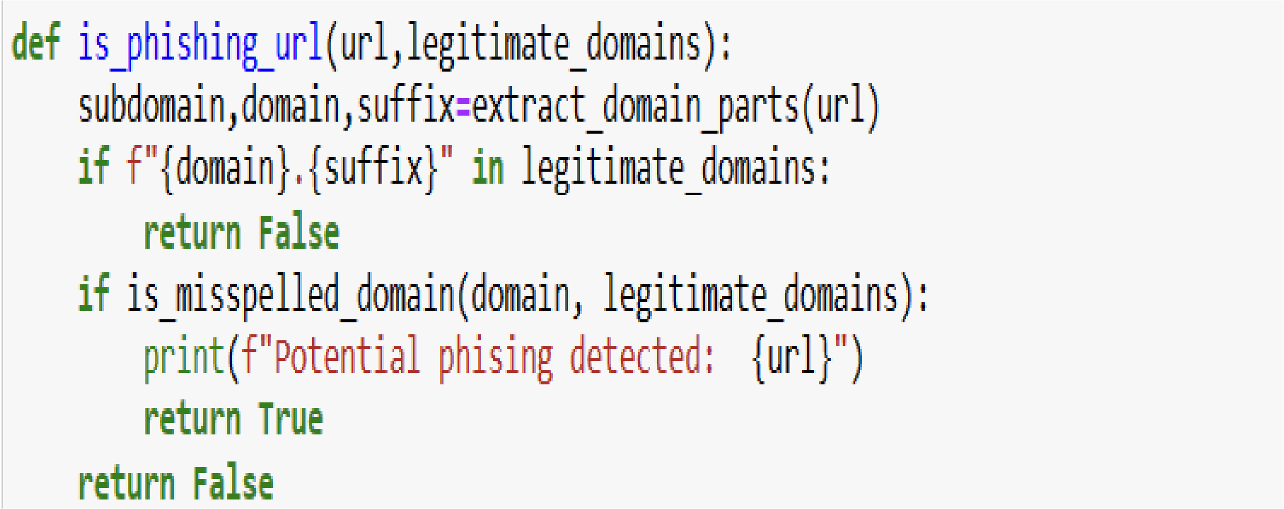
```
]
def extract_domain_parts(urls):
    extracted= tldextract.extract(url)
    return extracted.subdomain, extracted.domain, extracted.suffix
```

3. **Content Inspection**: The tool retrieves the content of the webpage pointed to by the URL and inspects it for signs of phishing, such as deceptive forms requesting sensitive information, suspicious pop-ups, or misleading content.



```
def is_misspelled_domain(domain,legitimate_domains,threshold=0.9):
    for legit_domain in legitimate_domains:
        similarity=lv.ratio(domain,legit_domain)
        if similarity>=threshold:
            return False
    return True
```

4. **Link Verification**: It verifies whether the URL redirects to a legitimate website or if it leads to a phishing page. This involves following redirects and comparing the final destination with known trustworthy sources.



```
def is_phishing_url(url,legitimate_domains):
    subdomain,domain,suffix=extract_domain_parts(url)
    if f"{domain}.{suffix}" in legitimate_domains:
        return False
    if is_misspelled_domain(domain, legitimate_domains):
        print(f"Potential phising detected:  {url}")
        return True
    return False
```

Code Explaination:-

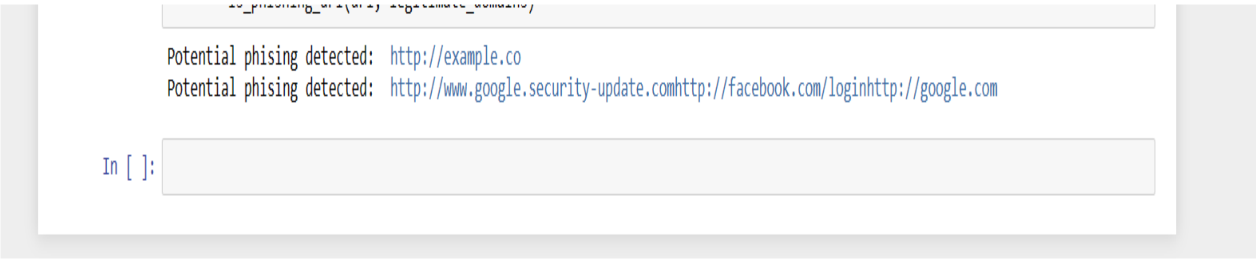
1. **Importing Libraries**:
    - **tldextract**: This library helps in extracting the domain parts from a URL.
    - **Levenshtein**: This library provides functions to calculate string similarity using Levenshtein distance.
2. **Initial Setup**:
    - **legitimate_domains**: This is a list containing known legitimate domain names.
    - **test_urls**: This is a list of URLs that we want to test for potential phishing.
3. **Function Definitions**:
    - **extract_domain_parts(urls)**: This function takes a URL as input, extracts its subdomain, domain, and suffix parts using **tldextract**, and returns them.
    - **is_misspelled_domain(domain, legitimate_domains, threshold=0.9)**: This function checks if the given domain is misspelled by comparing it with the known legitimate domains. It calculates the similarity ratio between the given domain and each legitimate domain using the Levenshtein distance algorithm. If the similarity ratio exceeds a certain threshold (default is 0.9), it considers the domain as misspelled.
    - **is_phishing_url(url, legitimate_domains)**: This function takes a URL and the list of legitimate domains as input. It extracts the domain part from the URL, checks if it matches any of the legitimate domains directly, and if not, it checks if it's a potential misspelled domain using

is_misspelled_domain(). If it detects a potential phishing URL, it prints a warning message.

4. **Main Execution**:

   - In the **if __name__ == '__main__':** block, it iterates over each test URL in the **test_urls** list and calls the **is_phishing_url()** function to check for potential phishing. If a URL is identified as potentially phishing, it prints a warning message.

Output:-



```
    is_phishing_url(url, legitimate_domains)

Potential phising detected:  http://example.co
Potential phising detected:  http://www.google.security-update.comhttp://facebook.com/loginhttp://google.com

In [ ]:
```

The final result of Phishing link detection will be return as result from the code.