Name : Khushi Ganesh Suvarna

Intern ID : 121

Homograph Detector Tool

Introduction

This tool is designed to detect homograph attacks by identifying visually deceptive characters (homoglyphs) used in domain names. These characters, often from scripts like Cyrillic or Greek, can be used by attackers to impersonate legitimate websites.

Problem Statement

Cybercriminals use similar-looking Unicode characters to create fake domains that appear identical to real ones, tricking users into visiting phishing or malicious sites. Detecting such characters manually is nearly impossible for most users.

Technology Used

- **Python 3**: Main programming language used to build the tool.
- **unicodedata module**: Python's built-in module used to analyze the Unicode name and script of each character.
- **Command Line Interface (CLI)**: Used for user input and output display.

Approach and Methodology

The tool scans each character in the domain and identifies its Unicode script (like Latin, Cyrillic, etc.).
It compares characters with a known homoglyph dictionary to detect look-alike letters.
If the domain uses both Latin and non-Latin scripts with suspicious characters, it flags it as a potential homograph attack.

Code Explanation

- **Importing unicodedata:**
  This module helps identify the Unicode name and script (e.g., LATIN, CYRILLIC) of each character in the domain.

- **homoglyphs dictionary:**
  Contains characters from other languages that look like English letters, used to detect suspicious replacements.
- **get_script(char) function:**
  Returns the script type (like LATIN or CYRILLIC) for a given character using its Unicode name.
- **detect_homograph(domain) function:**
  - Loops through each character in the domain
  - Stores the scripts used
  - Checks if any character is a known homoglyph
  - If the domain uses both Latin and non-Latin scripts and includes homoglyphs, it's marked suspicious
- **User Input & Output:**
  - Takes domain input from the user
  - Prints whether the domain is safe or suspicious based on the analysis
  - Shows which scripts are used and which characters are suspicious

Examples and Demonstrations

Sr.No	URL	Result	Explanation
1	google.com	Suspicious	The second 'o' is a Cyrillic 'o', not a Latin 'o'. Mixed script detected.
2	github.com	Safe	All characters are from the Latin script. No homoglyphs found.
3	amazon.com	Suspicious	Two Cyrillic 'a' characters used in place of Latin 'a'. Phishing possible.
4	microsoft.com	Suspicious	The letter 'o' is a Cyrillic o, not an English 'o'. Mixed-script domain.
5	facebook.com	Safe	No foreign characters used. Only Latin script present — domain is clean.

Real-World Applications

- **Phishing Protection:**
  Detects fake URLs that mimic real websites, helping prevent phishing attacks.
- **Browser Security Extensions:**
  Can be integrated into web browsers to warn users before visiting suspicious sites.
- **Email & Spam Filters:**
  Used to scan links in emails and block malicious or deceptive domains

.

Conclusion

The Homograph Detection Tool helps identify suspicious domain names that use deceptive Unicode characters (homoglyphs) to mimic trusted websites. By analyzing the scripts and characters used in a URL, the tool can detect potential homograph attacks and alert users. This simple Python-based solution adds an extra layer of protection against phishing and cyber fraud, making it a valuable addition to basic cybersecurity tools.