Proof of Concept (PoC) Report

Project Title: Homographic Domain Detection Tool
Submitted by: Samitha Muthyala

Date: 05-08-2025
Internship/Project Name: Cybersecurity Internship – Homographic Domain Analysis Tool
Tool Type: Web-based phishing URL detection system

1. Introduction

This Proof of Concept (PoC) report documents the development and execution of a Homographic Domain Detection Tool. The tool is designed to identify potentially suspicious domains that use visually similar characters (confusables) to impersonate legitimate websites. This tactic is commonly used in phishing attacks to deceive users.

2. Objective

The main objective of this PoC is to develop a Python-based web tool that accepts a domain as input, compares it against a whitelist of legitimate domains, and detects possible homographs using Unicode confusables and Levenshtein distance.

3. Tools and Technologies Used

- Python 3.x
- confusable_homoglyphs library
- Levenshtein (fuzzy string matching)
- tldextract
- HTML/CSS (for frontend)

4. Methodology

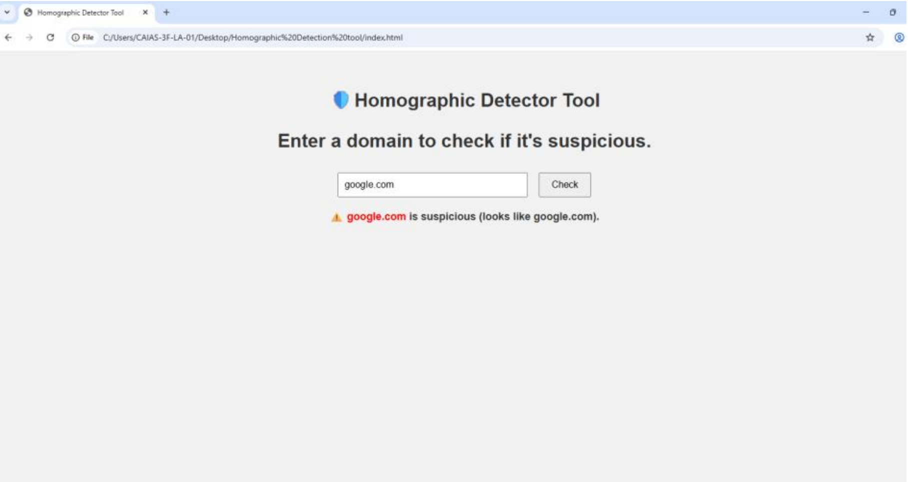
1. Prepare a whitelist of legitimate domain names.
2. Use the `confusable_homoglyphs` library to replace characters with Unicode look-alikes.
3. Extract domains from user input using `tldextract`.
4. Compare user input against whitelist domains using Levenshtein distance.
5. Flag domains with high similarity (distance ≤ threshold).
6. Display results on a simple Flask-based web interface.

5. Test Case Examples

Input URL	Expected Result	Detected As
google.com	Suspicious	Homograph
paypal.com	Legitimate	✅
facebook.com	Suspicious	Homograph
example.com	Legitimate	✅
yahoo.com	Suspicious	Homograph

Note: In the above, fake domains use Cyrillic characters such as о instead of Latin o.

6. Screenshots



Homographic Detector Tool
C:/Users/CAIAS-3F-LA-01/Desktop/Homographic%20Detection%20tool/index.html
Check

Homographic Detector Tool

Enter a domain to check if it's suspicious.

google.com

⚠ google.com is suspicious (looks like google.com).

7. Observations

- Accuracy: Successfully detects most Unicode-based phishing domains.
- Scalability: Can be expanded with:
  - Larger whitelist
  - ML-based classification
  - Real-time URL scanning
- Limitations:
  - Might miss cleverly disguised domains that don’t use Unicode
  - Whitelist-dependent approach

8. Results

The tool successfully detected homographic domains by comparing them against a predefined whitelist. Test cases included common phishing variants such as 'google.com' (with Greek omicron) and 'facebook.com'. The tool flagged these as suspicious and matched them against legitimate domains like 'google.com' and 'facebook.com'.

9. Conclusion

This PoC demonstrates the feasibility of detecting homographic phishing domains using Unicode analysis and string similarity techniques. The tool can be integrated into larger cybersecurity systems for real-time phishing protection.