**Name : Nikita Shukla**

**Intern ID : 134**

**Proof of Concept (PoC) – Homoglyph Domain Detection**

**1. Title**

Homoglyph Attack Detection Tool – Proof of Concept

**2. Objective**

The objective of this PoC is to demonstrate a Python-based detection mechanism that

identifies potentially malicious domain names or URLs that use homoglyph characters

(visually similar Unicode characters) to mimic legitimate domains (e.g., replacing the Latin

"a" with the Cyrillic "а").

**3. Background**

Homoglyph attacks, also known as IDN (Internationalized Domain Name) homograph

attacks, exploit the visual similarity between characters from different scripts (Latin,

Cyrillic, Greek, etc.) to create deceptive domain names.

Example:

 Legitimate: google.com

 Malicious: ɡoogle.com (Cyrillic small letter "ɡ")

Such domains are often used for phishing attacks, credential harvesting, or malware

delivery**.**

**4. Scope**

 Detect suspicious domains using Unicode normalization and homoglyph replacement.

 Compare normalized domains against a whitelist of known legitimate domains.

 Identify and flag domains that have high similarity scores with trusted domains.

**5. Methodology**

The PoC follows these steps:

**Step 1 – Input URL**

The user provides a URL suspected of being malicious.

**Step 2 – Domain Extraction**

Extracts the domain name from the URL using regex.

**Step 3 – Unicode Normalization**

Uses Python’s unicodedata module to normalize the text into NFKC form.

**Step 4 – Homoglyph Replacement**

Maps suspicious Unicode characters to their ASCII equivalents using a predefined

homoglyph dictionary.

**Step 5 – Similarity Check**

Uses difflib.SequenceMatcher to compare the normalized domain with whitelisted legitimate

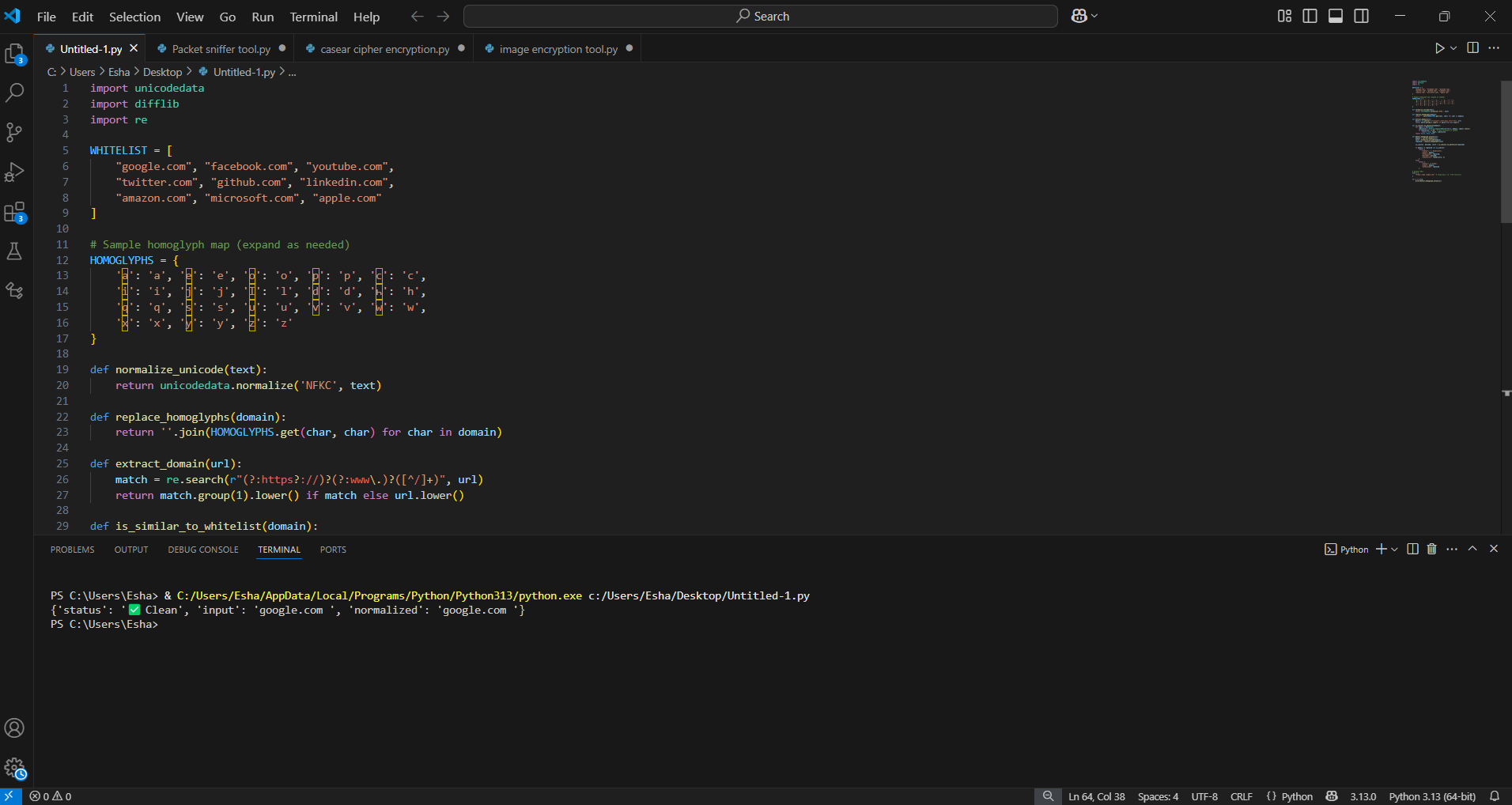
domains.

If the similarity score > 0.85, it’s flagged.

**Step 6 – Output**

Returns JSON-like results indicating whether the domain is Clean or Suspicious, along with

similarity score.



**7. Limitations**

 The homoglyph mapping dictionary is not exhaustive.

 Threshold-based similarity may lead to false positives or negatives.

 Does not actively block access — only detects.

**8. Conclusion**

This PoC successfully demonstrates the detection of homoglyph-based phishing domains.

Expanding the homoglyph dictionary and integrating this into a real-time security tool could

enhance web browsing safety