Homographic (Homoglyph) Detector

Name : Lavanya Chilukamari

Intern ID : 308

Objective: Develop a basic detection mechanism that identifies potentially malicious domain names or URLs that use homoglyphs (visually similar Unicode characters) to mimic legitimate domains (e.g., ".google.com" instead of "google.com").

What Is a Homograph Attack?

It’s when an attacker uses characters like:

- a (Cyrillic 'a') instead of a (Latin)
- l (Cyrillic small letter palochka) instead of l

Example: apple.com vs apple.com (the first is malicious).

Code :

```
import unicodedata

from urllib.parse import urlparse


def is_homograph(url: str) -> bool:

    try:

        parsed = urlparse(url)

        domain = parsed.netloc or parsed.path  # Handles cases without scheme


        # Check for non-ASCII characters

        if any(ord(char) > 127 for char in domain):

            print(f"[!] Suspicious: Domain contains non-ASCII characters: {domain}")


            # Try converting domain to punycode and back to detect changes

            try:

                punycode = domain.encode('idna').decode('ascii')

                restored = punycode.encode('ascii').decode('idna')

                if domain != restored:

                    print(f"[!] Domain mismatch after encoding/decoding: {domain} != {restored}")
                    return True
                except Exception as e:
                    print(f"[!] Error in IDNA encoding/decoding: {e}")
                    return True
            return False
        except Exception as e:
            print(f"[!] Failed to parse URL: {e}")
            return True

# Test examples
urls = [
    "https://apple.com",                  # Legitimate
    "https://apple.com",                  # Homograph using Cyrillic
    "http://www.google.com",
    "http://www.googłe.com",              # Homograph using Polish 'ł'
    "https://xn--pple-43d.com",           # Punycode of a suspicious domain
]

for url in urls:
    result = is_homograph(url)
    print(f"URL: {url} --> {'⚠️ Homograph Detected' if result else '✅ Safe'}\n")
```

Output :

```
URL: https://apple.com --> ✅ Safe
[!] Suspicious: Domain contains non-ASCII characters: apple.com
URL: https://apple.com --> ✅ Safe
URL: http://www.google.com --> ✅ Safe
[!] Suspicious: Domain contains non-ASCII characters: www.googłe.com
URL: http://www.googłe.com --> ✅ Safe
URL: https://xn--pple-43d.com --> ✅ Safe
```