CS5331 Web Security – SmellsPhishy (Chrome Extension for Phishing Detection)

Samantha Wong Shin Nee, Teo Ee Siang Lionel, Leong Peng Kuen, Quek Zhi Qin Daryl
School of Computing, National University of Singapore

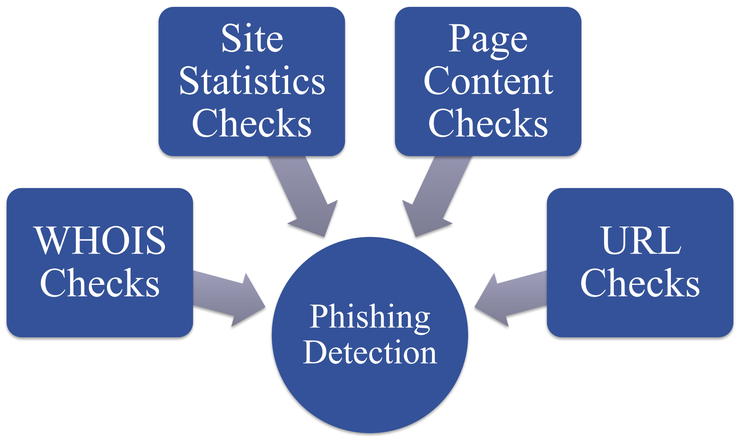
Introduction

- **Punycode (Homograph Attacks) –** Attackers can replace certain characters in existing URLs with Unicode character that looks exactly the same.
  - An attacker can host a site with the domain “google.com” using the Greek small letter (U+03BF) that looks like "o" and redirect to his own malicious webpage.
- **Redirects –** Long URLs are difficult to distribute and remember, hence shortened URLs are used e.g. bitly. Malicious redirects can be hidden in shortened URLs and used to disguise the underlying address. Redirects can also be triggered by embedded JavaScript code.

Objective

- Our project aims to warn users before they visit malicious websites unknowingly to defeat homograph attacks and malicious redirects.
- Demonstrate SmellsPhishy extension performing an analysis on webpage that the user is browsing to. If the webpage is likely to be malicious, the user will be prompted with a warning and asked to decided to proceed or cancel.

Logical Overview



Breakdown of Detection Techniques

Technique	Recommended Threshold	Source
URL Checks		
Top Level Domains	Blacklist	Parsing
Length of Base Domain	< 12 characters	Parsing
WHOIS Checks		
Domain Age	> 1 month	WHOIS Records
Domain Expiry	< 3 months	WHOIS Records
Registrant Verification	Top Google search result for registrant must be a page registered by registrant	WHOIS Records
Site Statistics Checks		
URL Page Ranking	+/- 10 ranks from possible identified original site	API (SimilarWeb)
URL Traffic Hits	+/- 50% traffic hits from possible identified original site	API (SimilarWeb)
Page Content Checks		
Forms		HTML Source

Detection Components

- **URL.** Analyzes the URL of the webpage being visited.
  - Domain Content.
  - Domain Length.
- **WHOIS.** Performs verification checks using data in the WHOIS record for the target URL’s domain.
  - Domain Age: ~96% of phishing sites are taken down within 1 month.
  - Domain Expiry: Legitimate domains are properly managed and renewed before they near expiry.
  - Registrant Verification: Registrant can be a parent/holdings company. They must be registrants for their own websites.
- **Site Statistics.** Perform comparison checks with possible identified original site provided with the following:
  - Page Ranking.
  - Traffic Hits.
- **Page Content.** Examines the target URL’s web page’s contents for suspicious elements.

Implementation

1. SmellsPhishy uses the Chrome API to intercept HTTP requests containing Punycode and HTTP responses containing redirects.
2. A Google search is conducted for the target URL’s page title. If URL of the top search result matches, this is considered to be a legitimate web page. Otherwise, it is put through the phishing detection engine with the possible identified original site.
3. The text elements of the URL are examined.
4. The web page is passed through the various WHOIS checks. These make use of a WHOIS service provider and retrieves the required field(s) for verification.
5. Site statistics are examined for the target URL and possible identified original site. Compare both websites for page ranking and traffic hits.
6. The target URL’s web page is parsed for suspicious elements.
7. Finally, the results of the phishing checks are displayed to the user. The user has to decide whether to proceed to visit the web page.

Challenges & Limitations

- WHOIS parsing is not trivial due to many different formats.
- Domain privacy strips availability of WHOIS data. This may soon become standard practice due to EU privacy regulations!
- Is using Google as a standard source of truth going to remain accurate going forward?
- May have limited text available on actual page with the clear use of images.
- Site statistics such as URL traffic hits might change if it is a very viral phish.

Conclusion and Future Work

- Many components involved in phishing detection.
- Gather more experimental data on live phishing sites to fine-tune threshold parameters.
- Add additional checks for visual components.
- Prompt user to input where they think they are going.
- Devise ways to infer the actual website from the page content.