Client Side Web Cloaking Detection

Anand Tripathi
atripath@cs.utah.edu

Kirk Webb
kwebb@cs.utah.edu

Anmol Vatsa
anvatsa@cs.utah.edu

September 15, 2017

1 Introduction

Web cloaking refers to the set of techniques that a web server uses to fingerprint incoming visitors in order to customize page content. An example can be, a website serving optimized pages for small screens. However, cloaking is often misused by miscreants to hide the true nature of malicious sites.

Invernizzi et al [1] measured the prevalence of cloaking on the web as well as its use by attackers to increase their search ranking and hide malicious content being served from web crawlers. The work also proposed a cloaking detection system but it's currently targeted towards search engines and security companies. One interesting insight of the study was that the content of cloaked websites retrieved from the client browser will be different from the one observed with a web crawler.

2 Proposal

The goal of this project is to investigate the possibility of detecting web cloaking on the client side. Specifically, we would like to build a browser plug-in that detects the use of cloaking by the website to block or alert the user.

We are going to use the aforementioned assumption that cloaked websites will serve different content to browser and web crawler by instantiating a parallel crawler request whenever the browser makes a new request. After retrieving the content from these requests, we can extract relevant content features and do a comparison [1] on the client-side to identify any major differences.

This project will include the following aspects:

- Lightweight client-side content classification.

  Our goal is to perform cloak detection on demand as a user browses without imposing cumbersome delays. We anticipate that this will involve lightweight machine learning techniques. We will explore trade-offs between detection accuracy and performance.

- Browser plug-in programming.

  None of us have prior experience with browser plug-in development, so there will be some amount of learning curve. We expect to study example implementations and learn tricks from open source plug-ins.

- Candidate web page harvesting and evaluation.

  We believe that we can acquire reasonable sets of pages to test against by surveying URLs contained in spam emails. We will also compose some of our own test web pages that attempt to present alternate views to web crawlers. We will run tests from multiple vantage points: On machines connecting through commodity home ISPs, from University machines, and from commercial cloud VMs.

References

[1] Invernizzi, Luca, et al. *"Cloak of Visibility: Detecting When Machines Browse A Different Web."* Security and Privacy (SP), 2016 IEEE Symposium on. IEEE, 2016..