11-5-2013

A Novel Defense Mechanism against Web Crawler Intrusion

Alireza Aghamohammadi

A Novel Defense Mechanism against Web Crawler Intrusion

by

Alireza Aghamohammadi

Dissertation

Submitted to the College of Technology

Eastern Michigan University

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY IN TECHNOLOGY

Area of Concentration: Information Assurance

Dissertation Committee:

Dr. Ali Eydgahi, Ph.D., (Chair)

Dr. Daniel Fields, Ph.D.,

Dr. Huei Lee, Ph.D.,

Dr. Alphonso Bellamy, Ph.D.

November 5, 2013

Ypsilanti, Michigan

Acknowledgments

This research would have not been possible without the assistance of the dedicated and outstanding committee members Dr. Ali Eydgahi (Chair), Dr. Daniel Fields, Dr. Huei Lee, and Dr. Alphonso Bellamy. The committee members' contributions and guidance were the most important factor in the successful completion of this study. I am especially grateful for Dr. Huei Lee's assistance and for making a computer lab available for this study. In addition, I also would like to acknowledge and thank my wonderful wife, parents, and sisters for their continuous support and understanding along the way. Their support and understanding have contributed to my success in so many ways, and I hope these few words will at least provide a glimpse of how much I appreciate their kindness and support through this Ph.D. journey and in my life.

Abstract

Web robots also known as crawlers or spiders are used by search engines, hackers and spammers to gather information about web pages. Timely detection and prevention of unwanted crawlers increases privacy and security of websites. In this research, a novel method to identify web crawlers is proposed to prevent unwanted crawler to access websites. The proposed method suggests a five-factor identification process to detect unwanted crawlers. This study provides the pretest and posttest results along with a systematic evaluation of web pages with the proposed identification technique versus web pages without the proposed identification process. An experiment was performed with repeated measures for two groups with each group containing ninety web pages. The outputs of the logistic regression analysis of treatment and control groups confirm the novel five-factor identification process as an effective mechanism to prevent unwanted web crawlers. This study concluded that the proposed five distinct identifier process is a very effective technique as demonstrated by a successful outcome.

Table of Contents

Chapter 1. Introduction

Introduction

The Internet has greatly impacted how information is created, shared, and accessed. It certainly has transformed how people, organizations, and governments function in terms of communication and collaboration. Some scholars and researchers even draw similarities between the Internet and other earlier inventions such as the printing press, telegraph, radio, telephone, fax, and how they all have changed the communication and lifestyle of many people around the globe (Feldman, 2002; Brown, 2009). In addition, the explosion of the Internet was so remarkable that it transformed the global economy, cultures, and society in terms of how people collaborate, share, and communicate, and still continues to evolve and impact culture, education, science, and so on (Divanna, 2003, p. 208). However, during the early days of the Internet, the process of adoption and use of the Internet was slow but steady until 1994 to 2000, at which point "the number of web hosts grew from 2.2 million to over 94 million" (Kogut, 2004). So, the Internet started simple and small but changed over time and grew as the result of new innovations in technology and in the number of users who started to use the Internet more often at home, work, and school with various devices such as smartphones or tablets.

The Internet started in 1969 in an experimental environment with only four computers connected to a very small communication network by agency of the U.S. Department of Defense called the Advanced Research Projects Agency (ARPA), in order to allow communications between researchers if a nuclear attack occurred (Nelson & Coleman, 2000). The technology used by ARPA was called TCP/IP, and even to this day, the Internet uses TCP/IP protocol to connect computers as the result of this ARPA successful project. Some

technology specialists and researchers even credit the Transmission Control Protocol (TCP)/ Internet Protocol (IP) model as the DoD standards, referring to its origin at the Department of Defense (Banzal, 2007). However, TCP/IP is not the only model for implementing protocol stacks; the Open Systems Interconnection (OSI) is another popular system that is used currently, and in terms of functionality, the layers of each model can be mapped to one another (Sathyan, 2010). Table 1 shows the layers of each model side by side in terms of functionality.

Table 1

OSI Model and TCP/IP Model

OSI Model	**TCP/IP Model**
Application layer	Application layer
Presentation layer	
Session layer	
Transport Layer	Transport Layer
Network Layer	Internet layer
Data Link layer	Network interface layer
Physical layer	

Both models provide similar functionalities, and there are not enough differences between the two models to examine each model separately for the purposes of this study. In this study, the TCP/IP is explained so a general understanding of the models is introduced to better understand the web infrastructure and system. The TCP/IP has four abstract layers (Steed, & Oliveira, 2009).

1. Application Layer

   This is where data are created and submitted to another computer. The main function of this layer is to access network functions. Applications use Internet Protocol (IP)

addresses and ports to communicate to each other. Port is simply a 16-bit unsigned integer such as 8080, and IP is the numerical address representation of a computer on a network.

2. Transport Layer

   This layer is responsible for managing and controlling the end-to-end communication for packets processing through a network. Transport layers primarily use two types of protocols: the User Datagram Protocol or UDP (a connectionless communication) and the Transmission Control Protocol or TCP (connection-oriented). Both protocols provide a process to communicate between client and host. UDP is faster but is less reliable in terms of how it communicates; TCP is more reliable.

3. Internet Layer

   This layer mainly is responsible for routing IP packets between computers. This layer creates, maintains, and ends network connections. IP packets provide information about the data communication process, as depicted in Table 2 (Steed & Oliveira, 2009).

   Table 2

   IP Packet

Bites	Packet				
0-31	Version	Header length	Type of service	Total length	
32-63	Identification			Flags	Frgament offset
64-95	Time to live	Protocol (upper layer)		header checksum	
96-127	Source address				
128-159	Destination address				
160-191	Options				
160+ or 192+	Data				

4. Network Access Layer

   The main function of this layer is to provide access for transmission, communication, and delivery of data across physical devices. For example, IEEE 802.11 or Ethernet are part of this layer.

The TCP/IP is a very powerful protocol used across the many computer networks and connects computers to the Internet. However, the infrastructure and architecture of the Web have multiple components at the application level, and that is the layer more visible to many Web users. There are two main computer network designs for implementing the application communication over TCP/IP. Below are brief descriptions of two main types of computer network architecture for implementing applications according to a book called *Networking Bible* (Sosinsky, 2009).

- Peer-to-Peer

  Each computer in a Peer-to-Peer network is called a node. Each node is considered an equal partner, and each node can act as a client and server by sharing resources. Furthermore, each node can have direct connection to another node, and there is no key management entity in the communication network. Many view this as a weakness because viruses or other harmful applications can easily get distributed to all nodes. For example, BitTorrent is a website based on Peer-to-Peer architecture.

- Client Server

  Client server is the most widely used application architecture on the web. Various applications and systems such e-mail systems, database systems, or simple web browsing on the Internet are all powered by client server architecture. Client server architecture has two main components: the client application and server application.

There is little limitation about this architecture except the client software should be able to communicate to the server application. The communication process between client and server is very similar to human communication because one has to initiate communication and the other person or entity has to respond. In a client-server environment, the client initiates the communication by sending a request to the server and in return the server will respond with a web page. Typically the request gets initiated by an individual who types the address in the browser, and the server will return the content of a web page or document as depicted in Figure 1.

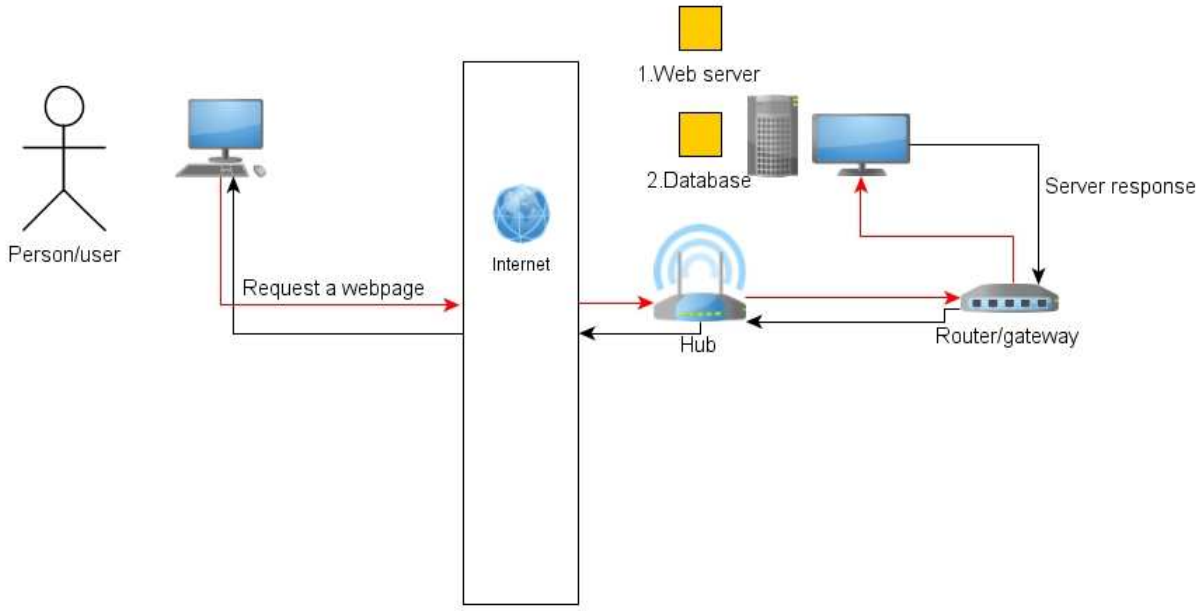


Figure 1. Client Server Application Architecture

The web and client server architecture worked well for the most part when the Internet started to grow, but it was very difficult to find and remember all the addresses and information on the web. According to a book called *SEO: Search Engine Optimization Bible*, the whole process of finding information on the Internet made it a “difficult” and “time consuming” experience (Ledford, 2007). Clearly, there was a need to find information faster

and not worry about the address or content of each web page. So web crawlers were created to help web users and search engines to create indexes of web pages and solve the problem of finding and remembering many web pages. In other words, the main goal of creating and using web crawlers was to address a human weakness because humans are simply much slower than computers when it comes to searching information. Web crawlers were created to find information and catalog web pages for search engines so that web users could easily find relevant information by using key words or phrases. The concept of indexing is very similar to the concept of creating indexes for books. For example, instead of going through every page in a book to find a specific keyword, indexes allow a faster way to find the specific content. The first person to implement the web crawling application with the concept of indexing was Matthew Gray in 1993 (Kuusisto, 2012).

Search engines have three main components. The first and most important part of the search engine is the crawler, which goes through web pages by reading every page and then following every link on each page. The second part of the search engine is indexes, which are the results of web crawlers and are simply a listing of the web pages that a web crawler reads. The third part of the search engine is a finder application with a distinct algorithm that goes to millions of web pages to find the best results for a searched key word or words. So search engines use crawlers to go to each web page one by one, automatically and consistently, first to catalog and index web pages and then to make the results of web crawlers searchable to all users (Stassopoulou & Dikaiakos, 2009). Web crawlers are critical because search engines cannot function without the web crawler's ability to gather information and catalog it as soon as it is created or modified on the web. Also, it is very important for a search engine to use the correct web crawler type to avoid storing huge

amounts of unused data in search engine databases; results and cataloged information from web crawlers get stored on a search engine's database so users can search these results by a key word or words. If an incorrect type of crawler is used, then terabytes of data would get stored on search engine servers without ever being used or accessed by any web users. There are two main types of web crawlers:

a. Generic Crawler

   The generic crawlers attempt to index and categorize pages regardless of subject or specific context (Govardhan, Narayana, & Premchand, 2009).

b. Focused Crawler

   Focused crawlers attempt to target a specific topic or subject. For example, the crawler may attempt to index and catalog any pages related to education, computers, or so on (Govardhan, Narayana, & Premchand, 2009). Furthermore, focused crawlers can even be subcategorized to topical (also known as the classic), semantic, and learning. The topical crawler accepts user input in the form of key words, starting with a set of URLs and then managing and controlling the results towards the pages that are more relevant to a given textual keyword (Menczer, Pant, & Srinivasan, 2004). Semantic crawlers function very similarly to the topical crawlers; however, the semantic crawlers start with some links but search and manage based on the semantics or context of given key words instead of crawling or searching for an exact key phrase (Ehrig & Maedche, 2003). For example, if given input is *education*, then the crawler will search for *school*, *universities*, and so on. Unlike the two previous types of focused crawlers, the learning crawlers are provided with training data and

will improve and learn methodology in order to find and target correct URLs or web pages (Batsakis, Petrakis, & Milios, 2009).

Clearly, the search engine crawlers have become very efficient in gathering information and analyzing the results, but all web crawlers are not created to gather information for search engines because they are also used by cyber-criminals, hackers, and spammers for "different types of unethical functions and activities such as automatic extraction of email and personal identification information as well as service attacks" (Sun, 2008). One of the current challenges of crawlers and web pages is to distinguish crawlers from other accesses in order to prevent undesirable web crawlers (Thelwall & Stuart, 2006; Zhong, 2010). Furthermore, researchers have created various documents about the misuse of web crawlers by other entities beside search engines, such as spammers, and the need to investigate how to identify web crawlers in order to prevent the unwanted web crawlers (Stassopoulou & Dikaiakos, 2009; Doran & Gokhale, 2011). So this study proposes a novel defense mechanism by using a five-factor identification process against web crawler intrusion in order to prevent unwanted web crawlers from gathering information and accessing web pages.

Statement of the Problem

Entering a web page via a crawler or robot to hack or steal information is unethical and creates privacy and security problems. Despite previous researchers' attempts to address the problem of identifying web crawlers versus humans to prevent misuse or theft of information on web pages, there is still a lack of information about how to effectively prevent all unwanted web crawlers from entering a web page without preventing humans and wanted web crawlers, such as Googlebot.

Nature and Significance of the Problem

The significance of being able to identify web crawlers to manage and prevent them has been documented by previous researchers (Lourenco & Belo, 2006; Tan & Kumar, 2002). In addition, there are multiple contributing factors supporting the significance of this problem. The followings are the main contributing factors:

First is the resource usage of web servers by the unwanted web crawlers. This challenge has been documented more recently as this continues to impact users, web administrator specialists, and software engineering in organizations. “A contemporary problem faced by site administrators is how to effectively manage crawler overload on dynamic web-sites” (Koehl & Wang, 2012, p. 171). The researchers found even though “crawlers only represent 6.68% of all requests, they consume an astonishing 31.76% of overall server processing time” (Koehl & Wang, 2012, p. 171). So even though there may not be a very high number of crawlers visiting each website, a few crawlers can impact server performance and processing in that servers and systems may not be able to process a high number of jobs or provide a prompt response to users. For example, if website resources are impacted, then a web page may not load or it may take a longer time to load. This impact on server performance as the result of an unwanted web crawler is not surprising because of how web crawlers function in a recursive or looping process. Web crawlers gather information by going into a recursive process for every hyperlink or link on each page until all the links on a given site are indexed. This recursive process is one of the main reasons why the server processing time is impacted by a limited number of web crawlers.

The second element contributing to the importance of preventing unwanted web crawlers is the security issue by using injection method. As a result of not being able to

prevent unwanted web crawlers, the websites are less secure and personal data are more accessible by criminals and those who want to steal the identity of others. There are various methods of using web crawlers to bypass security of websites such as the login page. For example, one approach involves the following: “a crawler requests a Web page and captures the response page. In the response page, it identifies input fields (e.g., HTML forms) which are filled and submitted with malicious inputs” (Shahriar & Zulkernine, 2012, p.15).

Third, current technology used to prevent web crawlers does not sufficiently protect web pages. A recent study found that more than 30% of active websites use Robots Exclusion Protocol (REP) to control web crawlers, but Robots Exclusion Protocol (REP) does not sufficiently manage web crawler’s access, and as the result there is a need to find a better solution (Giles, Sun, & Councill, 2010). The main reason REP does not protect and control web crawlers is that it functions only as an “unenforced advisory” mechanism (Giles, Sun, & Councill, 2010). The main challenge with REP is that web crawlers are expected to follow the robots.txt file rules which are set by the website owner or web page admin team, but the crawlers can simply ignore those rules if they want to.

Fourth, there is lack of new approaches to detect and prevent web crawlers because it is very difficult to identify and prevent web crawlers selectively without cloaking. According to an article entitled *Bots, Scrapers, and Other Unwanted Visitors to Your Web Site*, “there are technical solutions, but none is completely effective against a creative and determined bot designer” (Zabriskie, 2009). Also, Lourenco and Belo stated that “this is a widely recognized problem, there are few published papers in this particular area and techniques have not kept up with crawler evolving” (2006). Another reason why preventing web crawlers is challenging is that cloaking is discouraged and not permitted by various search engines.

Cloaking refers to a process whereby different content is displayed to different users or search engines, and since web pages are ranked based on their contents, search engines do not allow this (Wu & Davison, 2006).

The fifth element is the ability to prevent competitors from gaining access to marketing or pricing strategy which an online business may offer. One study documented that "many sites who advertise goods, services, and prices online desire protection against competitors that use crawlers to spy on their inventory" (Chandramouli & Gauch, 2007). This process of going to other websites to collect information via automated process or web crawlers is called *web scraping* and has recently created various legal challenges in courts (Watson, 2009). For example, the Momondo.com website provides price comparisons for cheap flights, but it never sought approval from Ryanair's flight (Compart, 2009). Another case was Southwest Airlines Co. v. Farechase, Inc., in which Southwest claimed that its terms of use prevent how Farechase was using web crawlers to do web scraping (Zabriskie, 2009). So it would be much easier to battle web scraping if there were a way to systematically and effectively prevent unwanted web crawlers.

Purpose and Objective(s) of the Study

The purpose of this study is to find a novel, systematic, and tested method to identify and prevent unwanted web crawlers accessing web pages without cloaking.

Research Questions and Hypotheses

Questions

The followings are the research questions for this study:

First, does the five-factor identification process which uses pass key, date, user agent, IP, and number of visits for the web server/page (allowed each day) significantly reduce unwanted web crawlers accessing web pages?

Second, does the five-factor identification process which uses pass key, date, user agent, IP, and number of visits for the web server/page (allowed each day) significantly reduce wanted web crawlers accessing web pages?

Hypotheses

The following hypotheses are defined for this study when comparing treatment groups and control groups. The treatment/intervention group is exposed to treatment and has five-factor identification. On the other hand, the control group was not exposed to the five-factor identification process at all.

Hypotheses Group A:

- ***H_0:*** There is no significant difference between treatment/intervention group and control group, in terms of wanted/valid web crawlers visits.
- ***H_1:*** There is a significant difference between treatment/intervention group and control group, in terms of wanted/valid web crawlers visits.

Hypotheses Group B:

- ***H_0:*** There is no significant difference between treatment/intervention group and control group, in terms of unwanted web crawlers visits.
- ***H_1:*** There is a significant difference between treatment/intervention group and control group, in terms of unwanted web crawlers visits.

Hypothesis Testing

Each of the hypotheses will be evaluated after the data analysis steps are completed and results are evaluated for accuracy and consistency. Hypothesis tests will be done based on the calculation of *p* valued. If the *p* value is greater than .05, then we do not reject the null hypothesis, but if the *p* value is less than or equal to .05, then we do reject H_0 in favor of H_1 hypothesis for each group. The concept of hypothesis testing using *p* value to compare against a pre-chosen alpha (usually $\alpha = 0.05$) to make decision about significance difference between two groups has been documented by various previous researchers and statistics authors (Schlotzhauer, 2009; Stephens, 2004).

Definition of Terms

Crawler, Robot, Spider, Scraper or Bot: Applications which go through Web pages automatically from one page to another page with a goal to retrieve information from Web pages (Stassopoulou & Dikaiakos, 2009).

Cloaking: A method or approach to show different web page content to different users (Lin, 2009). For example, when a person visits a news web page, the actual news would appear on the page, but if a crawler visits the same page, then different content is displayed.

Deep Web: The part of the web which is hidden to the common web crawlers because the content of those web pages is created dynamically or by dynamic web pages (Ke, Deng, Ng, & Lee, 2006). For example, a real estate website may require users to complete an online form about what type of home a potential online home buyer might be looking for, but the results are not displayed on the page until a person actually completes and submits the online form. These types of web contents are not visible to basic web crawlers and therefore they are often hidden as part of Deep Web.

Dynamic Web Pages: The pages which are created only when a query is submitted to server and the results are then created as a form of web page (Artail, & Fawaz, 2008).

Java: One of the well-known leading programming languages which has become the main language for web-based application and distributed computing (Taboada, Ramos, Exposito, Tourino, & Doallo, 2011).

Oracle Express: An Oracle software for database systems (Schrader et al., 2010).

Client and Server: *Client* in application and system context refers to anything that requests and consumes services. On the other hand, *server* is described as anything that provides services (Ruffer, Yen, & Lee, 1995).

Domain or Domain Name: Basically a conversion of numeric Internet Protocol or so-called IP address which provides a location for a computer on the Internet (Wang, 2006).

IP or IP address: A numeric number to uniquely identify hosts or computers on a network (Tsai, 2002).

Cached Information: Web browsers have a data storage location called cache, and when users visit various web pages, a copy of each page is stored into the cache location. This process of storing a web page on a user's computer helps to reduce the time to reload the page if the user decides to revisit the same page, because the page is already on user's computer and there is no need to go to the Internet to reload the same information (Branzburg, 2007).

HTML: Hyper Text Markup Language, which is a tag-based language created in a formatted way with heading, body, list and tables (Wise, 2007). The following is a sample of a very simple html code or tags (Wise, 2007):

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
</head>
<body>
</body>
</html>
```

Snippet: This word is usually used in the context of web page result. *Snippet* refers to a short description of a web page when a search results is displayed in a list. This information appears below or next to each link on the search result page (Google, 2012).

HTTP: Hypertext Transfer Protocol; it is the main application level protocol for the internet and it uses TCP/IP while it supports client-server communication in a stateless way.

Apache: A software organization which provides a lot of free open source software. It has various products including apache web server.

Tomcat: A web server used for Java application with servlets and Java server pages technologies.

Open Directory Project (ODP): "the largest, most comprehensive, and most widely distributed human-compiled taxonomy of links to websites, which makes extensive use of symbolic links" (Perugini, 2008, p. 910).

Md5 utility: A utility that uses md5 check sum algorithm for a given; it is used on most computers' operating systems (Rao & Vrudhula, 2007). This tool checks for integrity of

a downloaded file by comparing it against the remote md5 remote file checksum (Rao & Vrudhula, 2007).

Assumptions

It is assumed the five-factor identification process proposed in this study will be used as a way to reduce web crawlers' intrusions for business or government agencies only.

Limitations

The followings are the limitations of this study:

First, this study has time and budget constraints in terms of collecting and replicating real data used for web pages. This study used only 90 web pages for each group, and these web pages were hosted on web servers on a LAN (local area network) only. It is impossible to replicate all the web pages on World Wide Web or even purchase various domain names with dedicated servers to replicate more web pages.

Second, the proposed study is only for client server architecture and does not include peer-to-peer networks. Most applications created and built on the web are based on client server architecture (Sosinsky, 2009). So the five-factor identification approach does not provide a solution for a minority of web applications.

Summary

This chapter provided a brief overview of the Internet and how it began. This chapter also explained about the infrastructures of the Internet and various technology and models currently available and used. In addition, it provided an introduction about web crawlers and how this technology is used, including as a mechanism to index web pages by search engines. This chapter introduced the main topics for this research as it pertains to web crawlers and described the challenges with using web crawlers by focusing on the misuse of

web crawlers for hacking and gathering information. A statement of the problem and the nature and significance of using web crawlers' intrusion were described. In addition, the purpose of this study, its justification, significance, and research questions, along with hypotheses, were stated and explained. The next chapter will elaborate in detail about background and literature review pertaining to web crawlers and several significant studies about web crawlers and earlier works by previous researchers.

Chapter 2. Background and Review of Literature

Introduction

This chapter provides background about web crawlers and examines various literature pertaining to web crawlers. The previous studies are reviewed to better understand the web crawler's functionality and use for gathering information and analysis. Furthermore, other studies and related works have been published about how to identify web crawlers and how previous researchers and scholars have attempted to address web crawlers' identification and prevention problem. This chapter explains various types of web crawlers to better understand and address the challenge of preventing unwanted web crawlers. Also, one of the main goals of this chapter was to document solutions and findings of previous research related to this study to confirm that this study does not replicate or propose the previous researchers' solutions for identifying and preventing web crawlers. The previous literature focused on Robots Exclusion Protocol, caching and performance algorithm, ethical aspects of crawlers, web crawler detection and cloaking, and deep web and crawler search. In addition, some studies were very distinct in terms of topic. Those studies which could not be categorized as a group are explained under miscellaneous studies.

Background

As briefly explained in Chapter I, the Internet and its content has changed since the early days when it evolved at the US Department of Defense to a new tool for education entities and organizations to where people publish and share their ideas and thoughts (Mowery & Simcoe, 2002). However, one of the main differences between the early days of the Internet compared to today is the number of web pages. For example, in 2000, "Web consists of approximately 2.5 billion documents, up from 1 billion pages at the beginning of

the year, with a rate of growth of 7.3 million pages per day" (University of California Berkeley, 2000). In the early days of the Internet, various web pages were created but there were far fewer people and organizations online. As the result of fewer people online, fewer web pages and content were created online compared to today's so-called big data. Big data is the enormous amount of data created on the Internet by social media web sites and Internet transactions. Big data is creating many technical challenges to manage processes and complete algorithms because it is difficult to do analytics or perform computing tasks on huge amount of data quickly (Chiang, Goes, & Stohr, 2012). For example, it has become more challenging for any online service providers such as search engines or social network organizations to collect and process various data on the Internet by using web crawlers because massive amounts of data are getting created on the web by Facebook, Twitter, Tumblr, Pinterest, and Reddit. Web crawlers play a vital role in processing data on the Internet, as described in Chapter I. However, there are complicated challenges with web crawlers too because there are various stakeholders of web crawlers each having their own view about web crawlers usage. In order to better understand background and current web crawlers, one has to examine the stakeholder's perspective to be able to provide a comprehensive solution for all stakeholder holders. There are four groups of stakeholders when it comes to web crawlers. The first and most obvious group are the search engine organizations. As explained earlier, search engine organizations are very interested in the field of information retrieval, and they use web crawlers to gather today's big data. Web crawlers used by search engines are becoming more efficient in terms of processing data, and they are usually used to automatically scan the Internet and websites for indexing context analysis. For example, Googlebot by Google, Slurp by Yahoo or bingbot, adidxbot, msnbot

by Microsoft crawlers are web crawlers created and supported by search engines with a goal to index web pages (Google, 2012; Yahoo, 2011; Microsoft, 2012).

The second group of stakeholders is the web users. Web users go to search engines and use the data collected by web crawlers to find the information they are searching for online. The third group is the website owners or organizations which web crawlers go to and collect data, and the fourth group is the criminals and those who misuse web crawlers for collecting personal data such as emails. Criminals, spammers, hackers, and marketing organizations even use web crawlers despite knowing that collecting and accessing a web page by using web crawlers without obtaining permission has been viewed as an invasion of privacy and intrusion (Giles, Sun, & Councill, 2010). Previous researchers have even proposed solutions such as implementing robot.txt, also known as Robots Exclusion Protocol, to exclude pages or limit web crawlers' access, but various studies show that this protocol is not enforced and is ineffective (Sun, Zhuang, & Giles, 2007; Kolay, D'Alberto, Dasdan, & Bhattacharjee, 2008). The details of robot.txt and its functionality will be described in detail under Review of Literature and Related Works section of this study.

Preventing all crawlers access to web sites is possible but not practical because search engines robots/crawlers need to have access to web pages in order to index web contents and make them available to public through search engines (Madhavan, Ko, Kot, Ganapathy, Rasmussen, & Halevy, 2008). Well known search engines such as Google and Yahoo were not very strict about web pages few years ago, and they even allowed cloaking or displaying different web pages for humans and web crawlers (Wu & Davison, 2006). In fact various types of cloaking, including syntactic and semantic, were even mentioned in documents as early as 2006. Syntactic cloaking is simply the way in which two different contents are

presented to real users versus a web crawler; on the other hand, semantic cloaking is a method which presents various types of content to web crawlers in a meaningful way to increase the ranking of a web page in a search engine (Wu & Davison, 2006). However, more recently, some search engines such as Google and Yahoo no longer allow cloaking, and they clearly specify this in their terms of use guideline pages because search engines cannot accurately return results if a different web page is indexed by a web crawler as the result of cloaking (Google, 2012; Yahoo, 2011). Therefore, the current problem is being able to identify and prevent unwanted crawlers to access web sites without cloaking.

Review of Literature and Related Works

Web crawlers have previously been studied by other researchers. However, this study is different from previous studies because they focused on different topics, approaches, and solutions related to web crawlers. Previous studies can be categorized into the following topics:

a. Robots Exclusion Protocol, META Tags and X-Robots-Tag

Robot Exclusion or Robot.txt is a protocol to prevent web crawlers entering web pages or to have limited access to web pages. Web administrators or Web engineers use a file called robots.txt "to indicate to visiting robots which parts of their sites should not be visited by the robot" (Stassopoulou & Dikaiakos, 2009, p. 265). Here is an example of robots.txt file (Mao, & Herley, 2011):

User-agent: msnbot

Disallow: /private

User-agent: *

Disallow: /

The above lines indicate that web crawler msnbot is allowed to visit all web pages and folders except the private folder, while other web crawlers are not allowed to visit any pages or folders. In addition to robots.txt there is a tagging mechanism which can be used as part of a web page. These tags are referred to as META tags and can be used for various purposes including defining crawlers' access. For example, META tag for defining crawlers' permission may appear this way in html code for a web page:

```
<html>
<head>
<meta name="robots" content="noindex, nofollow">
<meta name="description" content="page description.">
<title>
The title of a web page
</title>
</head>
<body>
```

The "meta name=robots" in above code indicates the tag is intended to be used for defining the mechanism for crawler (Yalcin & Kose, 2010). If a word "robots" is used, then it also implicitly covers all robots, but if a page is concerned with only one specific crawler, then the specific name, such as googlebot, will be mentioned (Yalcin & Kose, 2010). In addition to the name of META tags, there are seven content types which can be added and separated by a comma as described below (Google, 2012):

1. Sometimes the pages should not be indexed or archived, so the word *Noindex* will be used to indicate to crawlers that they should index or archive a web page.

2. It is important to somehow prevent crawlers from crawling and indexing other pages which are linked to a given page; for these types of scenarios, the word *Nofollow* will be used.
3. When search results are returned, snippet information appears next to or below the links on search engine results. In order to prevent and hide snippet information, the word *nosnippet* can be used.
4. Some crawlers use Open Directory Project to display information for titles or snippets. To prevent crawlers from using Open Directory Project information, the word *noodp* is placed in the content.
5. *Noarchive* is one way to propose to crawlers that the cached link should not be displayed.
6. Sometimes web servers or web pages want to stop crawlers from indexing their pages after a given date and time. So the key word *unavailable_after* can be used to communicate to crawlers to stop indexing after a given date and time.
7. *Noimageindex* simply hides images of a web page from indexing.

So it is clear that there are various combinations of words that can be used to configure META tags. Furthermore, there are some ad hoc methods by search engine crawlers that can be used as a way to define crawlers' behavior on a given page. For example, Google even allows the use of X-Robots-Tag. The X-Robots-Tag is a simply an HTTP response tag which can be configured via httpd.con and .htaccess files for Apache-based web servers such as Tomcat (Google, 2012). Below are sample lines that can be added to .htaccess or httpd.conf file to manage web crawlers (Google, 2012):

<Files ~ "\.pdf$">

  Header set X-Robots-Tag "noindex, nofollow"

</Files>

X-Robots-Tag in .htaccess or httpd.conf works very similarly to META tag and describes how the owner of web pages prefers the crawlers to process or not process data on web pages.

The main weakness of protocols explained earlier is lack of enforcement. This means Robot Exclusion Protocol cannot guarantee or prevent unwanted crawlers because it works only if the web crawler is programmed to follow the guidelines in the robot.txt file (Giles, Sun, & Councill, 2010). Various previous studies focused on Robots Exclusion Protocol (REP) and how it is implemented and performs (Kolay, D'Alberto, Dasdan, & Bhattacharjee, 2008; Sun, Zhuang, & Giles, 2007). For example, one study focused on how robots.txt is used, but some web pages have a favorable or unfavorable bias against web crawlers based on the rules defined in robots.txt (Kolay, D'Alberto, Dasdan, & Bhattacharjee, 2008). Another important study related to Robot Exclusion Protocol was completed to see how robots.txt is being used for various sectors such as government, businesses, and education-related web pages (Sun, Zhuang, & Giles, 2007). Also, there were some earlier studies suggesting that the use of web crawlers will have limitations given the growing size of the web (Koster, 1995). Koster's study is one of the earliest studies explaining about how to use and implement robots.txt (Koster, 1995). Among various research, only a few pertained to measuring web crawlers and respecting the robot.txt standards (Giles, Sun, & Councill, 2010).

b. Web Caching and Performance Optimization

In addition to Robots Exclusion Protocol, META Tags and X-Robots-Tag studies, there are many previous studies focused on the topics of caching and performance algorithms of web crawlers. For example, one of the important studies pertaining to caching looked at how web server caching is used (Giles, Sun, & Councill, 2010). This study focused on the rate of change and caching of web contents and found that only 22% of resources were accessed more than once (Douglis, Feldmann, Krishnamurthy, & Mogul, 1997). The result of this research helped to better understand the use of various web resources in context of caching for status 200 and status 304 only. Status 200 is the standard http return code returned to a user's web browser from the web server when a request for downloading a web page has succeeded (Krishnamurthy, Mogul, & Kristol, 1999). Status 304 is when the requested page matches to the last requested page and the resource has not been changed since last requested (Krishnamurthy et al., 1999). The study called *Rate of Change and other Metrics:a Live Study of the World Wide Web* measured the four following factors when examining web request responses with caching challenging on the server and client side such as web browsers (Douglis et al., 1997):

Request Times: The number of requests from client to server and the time between every single request which is submitted to the server.

Modification Times: These data were extracted from the header information of http response. Those responses which returned 200 http codes had the last modification but for those responses which the page was dynamically created.

Age: This was calculated by the time difference between request time and last modified time. However, the study used 0 for those where data were not available.

Modification intervals: These were calculated by comparing the two consecutive responses based on modification time. However, in order to detect modification, the bodies of response were compared to see any modifications.

This study of caching performance concluded many resources change and "the frequency of access, age since last modified, and frequency of modification depend on several factors, especially content type and top-level domain, but not size" (Douglis, Feldmann, Krishnamurthy, & Mogul, 1997). Another study of this type, with the goal to investigate caching performance but focused on characterizing Web resources, server response, and Web caching behavior is called *Towards a Better Understanding of Web Resources and Server Responses for Improved Caching* (Wills & Mikhailov, 1999). Furthermore, this study looked at "characteristics of embedded images," "Changes to HTML resources," and "Cookies" in terms of rate of change (Wills & Mikhailov, 1999). The author's study is distinguished from previous studies by identifying two main points. First, the study used a method to analyze web-caching changes in a controlled way (Wills & Mikhailov, 1999). Second, the research also focused on web-caching-related issues in order to understand the request and response and caching (Wills & Mikhailov, 1999). However, to summarize the main direction of this study in terms of investigation, it is accurate to address the following as described by the authors (Wills & Mikhailov, 1999).

The study monitored the web resources to see the frequency change of the resources. The authors claimed other studies used the same technique previously, but their work created

an environment that allowed controlling the requests sent to server and testing those changes using MD5 checksum algorithm.

The second point of the study was about the availability and accuracy of cache validation processing. The researcher used the headers' returned information such as last modification time, size, and entity tags. The existence of last modification time was critical since the test for this research used the GET method of http protocol request.

The third point highlighted in the study was about how images and other included resources changed when compared to HTML code. According to the authors, previous investigation by other researchers had suggested that the rate of change for images was different from other components of a web page such as URL or text.

The fourth point this research looked into was the predictability and locality of changes. This is critical because dynamic web pages whose contents are generated as the result of submitting a query to a web host are impacted by caching particular components such as images.

The fifth point the authors investigated was to see how servers respond to different types of requests. One approach documented by authors specifically indicated the use of cookies to see if those cookies are returned to servers.

The data collection for the study called *Towards a Better Understanding of Web Resources and Server Responses for Improved Caching* only used the GET method of HTTP for each URLS in their test set. Each test set included at most 19 URLs (Wills & Mikhailov, 1999). The researchers found that "there is potential to reuse more cached resources than is currently being realized due to inaccurate and nonexistent directives" (Wills & Mikhailov, 1999). In addition to previous studies, there were some studies which attempted to focus on

optimization and performance improvement of web crawlers instead of caching only (Edwards, McCurley, & Tomlin, 2001; Cho & Garcia-Molina, 2003; Lee, Leonard, Wang, & Loguinov, 2008; Cai,Yang, Lai, Wang, & Zhang, 2008). For example, one study looked at how to create a web crawler with an optimized model that matches a strategy of web crawler while allowing improved process for controlling the results (Edwards, McCurley, & Tomlin, 2001). The researchers for the study described three approaches of crawling. The first approach is a process where all pages are crawled systematically and in the same order repeatedly (Edwards, McCurley, & Tomlin, 2001). The second approach is the random order, in which all pages still are crawled by a crawler but in a random, not sequential way (Edwards, McCurley, & Tomlin, 2001). The third approach is called purely random and it suggested a more ad hoc approach where some pages are crawled frequently but some pages are never crawled (Edwards, McCurley, & Tomlin, 2001). This study created three experiments with three strategies to replicate Web contents and crawling to test their mathematical models (Edwards, McCurley, & Tomlin, 2001). It attempted to examine three strategies to minimize the total number of obsolete pages which are explained in the following (Edwards, McCurley, & Tomlin, 2001).

The first strategy is a testing environment where equal weight is given to each period of each web crawling cycle.

The second strategy is a way to have the last time period with weight =1 or also known as the total weight, while other times zero weight would be used. The goal of this approach is to minimize the obsolete pages only for the last time period of crawling cycle.

The third strategy occurs in an environment where the last time periods would have higher weights, while at other times the weight would be set to low. The approach is very

similar to the second strategy, but the difference is that the goal in this strategy is to minimize the obsolete pages in all time periods and not just the last time period of crawling cycle.

The results from the study called *An adaptive model for optimizing performance of an incremental web crawler* suggested that an efficient crawling strategy can be used for incremental crawlers without making any general assumptions about how often web pages change, but the actual web-crawling cycles need to be used instead (Edwards, McCurley, & Tomlin, 2001). Also this study described the model the researchers provided in an adoptive and useful way because within each cycle of crawling, it managed the URL queues over a component time of each period and between the cycles during which the data gets changed for the best possible results. The experiment provided information that suggests crawling should be done only once during each cycle, and it also updated the next web crawling cycle for the best results (Edwards, McCurley, & Tomlin, 2001).

Another study of this type which contributed to the field of web-caching and performance improvement but more specifically effective refresh policies for web crawlers was by two researchers from the University of California and Stanford University (Cho & Garcia-Molina, 2003). The authors of this study highlight one of the main challenges that many search engine providers face, which is the lack of ability to easily obtain a fresh copy of web pages since crawling all the pages is very expensive in terms of processing, and when web contents change, the crawlers or search engines are not notified by web pages (Cho & Garcia-Molina, 2003). The research provided very detailed information, which can be summarized in the following (Cho & Garcia-Molina, 2003).

First, the study provided framework about how to address synchronization challenges by examining freshness (which was defined as the more up-to-date element present in a given

dataset), and age was documented as the interval time between the last update date and current date (Cho & Garcia-Molina, 2003).

Second, the research provided insight to some of the synchronization policies which might perform poorly but are appealing because of their simplicity. The study pointed out the dimensions of the synchronization process in terms of synchronization frequency and allocation. Synchronization frequency refers to how frequently local databases are synchronized with the actual web pages, and resource allocation determines how many elements to synchronize per unit of interval and how frequently to synchronize each individual element.

Third, the study proposed a new synchronization process as a way to have better results in terms of freshness by orders. It is important to point out the policy recommended took into consideration the rate of change for web pages and the importance of changes for web pages given.

Fourth, the authors validated their experiment and the data they gathered from 270 websites. Also, the study examined how effective different methods are by using Poisson process. Poisson process is used to create models based on the real world, but the problem or environment should be sequential events which happen randomly and independently of one another within a fixed rate of time.

The study of *Effective Page Refresh Policies for Web Crawlers* found proportional-synchronization policy does work when it comes to using them for real world problems because the age of proportional policy was 93 times worse than optimal policy (Cho & Garcia-Molina, 2003). A more recent study about performance improvement was done by the Department of Computer Science, Texas A&M University, for the effective and efficient

processing of many web pages. This study focused on how to go about improving algorithm for downloading many web pages, and along the way it documented the following important challenges faced when crawling many web pages (Lee, Leonard, Wang, & Loguinov, 2008):

The first and most obvious challenge described by the researcher was processing and verifying distinct URLS without violating Robots Exclusion Protocol. This checking process becomes very time-consuming and creates a bottleneck.

The second challenge was the many pages which should not get processed as result of spam. Spam is not just a form of email; there are many web pages which have many target URLs or links in order to increase the target URLs ranking. So the challenge according to the authors is to implement an algorithm for Spam Tracking and Avoidance through Reputation to allow certain number of pages for each domain and subdomain.

The third processing problem is how to prevent live locks for processing URLs that go over their limits. For example, rescanning the same links created a just a little new information but added a huge overhead for reprocessing.

The study ran the crawler for 41.27 days; however, the main weakness of this research was that the proposed algorithm for their experiment excluded non-HTML pages, HTTP errors, and redirects, and only included the http error code 200, which is not a correct reflection of real search engine crawling (Lee, Leonard, Wang, & Loguinov, 2008). However, the study did propose a new algorithm for improving performance of web crawlers.

In order to tackle the performance and crawling efficiency problems, some studies proposed a targeted web crawler instead of trying to catalog and index all the web pages. This goal of the targeted crawling approach is to narrow the number of crawling web pages

by just focusing on a given area of interest. Some studies proposed web crawling based on key words or topics (Radhakishan, Farook, & Selvakumar, 2010; Kumar & Vig, 2009; Menczer, Pant, Srinivasan, & Ruiz, 2001; Mali, S., & Meshram, B.B., 2011). Among various solutions to address this performance and crawling problem, one study had a really interesting and useful solution compared to the typical crawling algorithm (Radhakishan et al., 2010). The authors proposed not to "archive the entire site in order to check for the presence of some word in its entire domain. This is highly inefficient and a lot of storage space is wasted in this process" (Radhakishan et al., 2010). The main advantage of this approach of is eliminating the need to archive files and web pages; this reduces the number of servers needed to store the files and web pages. In addition, it would be very cost-effective in terms of maintenance of software and hardware. However, the drawback is the lack of efficiency because when crawling across millions of web pages, it would be impossible to bring results back to users in seconds or milliseconds despite the cost saving provided by the study called *CRAYSE: Design and Implementation of Efficient Text Search Algorithm in a Web Crawler.* It would be very challenging to implement a real search engine without a data warehouse to store web pages unless the search is on only one domain. So if the approach proposed by Radhakishan, Farook, and Selvakumar is implemented for one domain name, then the process may bring positive results, but the process of searching the web may not be very practical or fast. In addition to Radhakishan, Farook, and Selvakumar's research, there were other studies that also attempted to take a different approach in terms of targeted crawling method. For example, one group of researchers examined a focused crawling approach in order to save time by just targeting relevant pages, which requires indexing instead of attempting to index many web pages without a specific target set. The researchers

described how difficult and challenging it is to find relevant information on the web as the result of growing information from web pages, servers, and documents (Kumar & Vig, 2009). In addition, the study pointed out that the information on the web is changing rapidly, and there is a need to avoid irrelevant information when crawling in order to better analyze and process data for search engines (Kumar & Vig, 2009).

In addition to previous work by Kumar and Vig, one study even focused on an algorithm with discovering URLs through user feedback (Bai, Cambazoglu, & Junqueira, 2011). The following were explained in the study as the drawbacks of current focused crawling based system (Kumar & Vig, 2009):

- First, lack of efficient relevance scoring process and tunneling mechanism (i.e., the process to find the relevant web pages from none relevant pages from given a page) has contributed to some of the weakness of focused crawling.
- Second, the focused crawlers only perform syntactic matching by simply finding a key work match from a user's input on the web pages. This is too simplistic and often returns inaccurate and irrelevant information.
- Third, query matching and scoring algorithm is flawed because it completely disregards the context of keywords.
- Fourth, there is an inability to understand the content of web pages and documents in order to find correct results for users when an input is provided to be processed.

The study provided a very high level architecture as far as what the "Context Ontology Rule Enhanced" would need, but few specifics were provided in terms of implementation process and technology (Kumar & Vig, 2009). The study documented an effective method for identifying more accurate focused crawling by using tables able to store

the importance of each key term's occurrence (Kumar & Vig, 2009). If the key term's occurrence and importance are stored, then they can be used when a query is submitted to a server. The stored values in tables will help to identify the key word importance relevant to each web page (Kumar & Vig, 2009).

c. Ethical Aspect of Autonomous Web Agents and Web Crawlers

The impact and role of ethics has been an important topic in the computer field from network programing to software security and hacking. However, the topic is a very complicated area of study, and various studies have been done previously to address challenges such as privacy. Also, there are many community-based needs pertaining to ethical issues and computers, including the web crawling or internet-based ethical questions. For example, according to a study called *Towards Community Standards for Ethical Behavior in Computer Security Research*, there are many questions which need answers, such as is it ok to break a computer network in order to demonstrate to others that the existing protocols do not work well? Or is it ok to deceive users in order to understand how some attackers deceive users (Dittrich, Bailey, & Dietrich, 2009)? The authors of a study proposed a community-based solution and the need to explore various existing ethical computer challenges such as various frameworks for security research (Dittrich, Bailey, & Dietrich, 2009). However, one of the most important cases which this study examined was about P2P and Botnets functioning as command and control servers (Dittrich, Bailey, & Dietrich, 2009). This study documented the use of web crawlers to take advantage of P2P algorithm (Dittrich, Bailey, & Dietrich, 2009). However, there were other researchers who examined the issue of ethical behavior related to web crawlers and autonomous software robots at a deeper level. For example, one study focused on web services as the autonomous

software agents and ethical challenge pertaining to use of autonomous software agents (Gangadharan & Pretorius, 2010). A research article called *Towards an Ethical Analysis of the W3C Web Services Architecture Model* examined whether technology such as web services should be subjected to ethical analysis and Floridi's theory (Gangadharan & Pretorius, 2010). The main area which this study elaborated includes the following (Gangadharan & Pretorius, 2010).

First, the researchers described the existing web services' architecture including Message Oriented Model (MOM), Resource Oriented Model (ROM), Service Oriented Model (SOM) and Policy Model.

The second point which was examined was Computer Ethics and Ethical Theories including early views about how to address ethical dilemmas about computer/human interaction.

Third, Floridi's Information Ethics, which explains how the autonomous agents, including web crawlers, could be viewed in terms of Level of Abstraction (LoA), were studied. Furthermore, each LoA is composed of moral agents and moral patients. The moral agents are any entities that can harm or benefit.

The fourth area examined was about applied analysis of Floridi's theory to web services in terms of interaction exchange or message communication between requester agent and provider manager.

The study concluded that technologies that function as autonomous software agents should be under examination for ethical challenges (Gangadharan & Pretorius, 2010). Also, the study found that by using Floridi's theory is possible to categorize web service components in terms of moral agent and moral patients and its rule but also recognized that

this theory may not be fully applied to some cases (Gangadharan & Pretorius, 2010). Even though the study by Gangadharan and Pretorius indirectly categorized web crawlers as software agents and elaborated on them a little, the main focus of study was autonomous software agents in general in computing environment with web services (Gangadharan & Pretorius, 2010). In addition, the study failed to address some of the main challenges raised by use of web crawlers as web agents, such as how web crawlers should interact with web pages when the owner of web page does not explicitly forbid the web crawlers access but does not want the information on his or her page to be gathered for marketing purposes. The topic of web crawls agent and ethical issues related to such applications were more directly studied in a research paper called *Ethical Web Agents,* which brought attention to the fact that the use of web agents such as spiders provides value to web users, but there is a great need to pay attention to not only the technical aspect of improving web crawlers but also the ethical challenges these agents have introduced to humans (Eichmann, 1995). The study explored the following areas (Eichmann, 1995).

First, intelligent software agents and web spiders were reviewed in terms of historical aspect and functionality impact. Also the issue of relationships between agents was also briefly examined along with how poorly designed of spiders can impact the overall network performance.

Second, a rationale for creating agents for the web was studied. For example, the distinction between hyper texting navigation and browsing experience was explained, including how people prefer browsing experience and how building it requires web service infrastructures.

Third, the concept of ethics related to the web was described. For example, Koster's guidelines for robot writers were examined and explained. Furthermore, the difficulty working with robot exclusion standards was supported by the fact that such a protocol does not force any limitation on spider agents.

Fourth, the problem about still facing many unresolved issues with use of web crawlers was described and examined. For example, what are virtual neighborhoods of information that can be created while managing the generated traffic by robots?

This study did a great job by providing the basis and most challenging aspects of dealing with ethical issues related to web crawlers (Eichmann, 1995). In addition to Eichmann's study, a more recent study with extensive details was completed about crawlers' regulations and behavior on the web in context of bias and ethicality measurement (Sun, 2008). This research covered the following points pertaining to web crawlers (Sun, 2008):

First, the thesis provided comprehensive information about web crawlers' behavior and functionality, including how crawlers gather information. In addition, breadth first search (BFS) and depth first (DFS) search were explored along with focused crawling.

Second, quantitative metrics and models were presented to measure web crawlers' biases and ethics. So various models including binary, probabilistic, relative, and cost model were explained.

Third, a detailed and complete survey of robots exclusion protocol was provided which confirmed that more than 30% of web pages use Robots.txt to manage crawlers' access. The study also found that many web servers were incorrectly using the Robots.txt standards because the implementation of Robots.txt by the web crawlers is dependent on how the web crawler is created to process robots.txt standards.

Fourth, the ethical issues of web crawlers were investigated in detail in terms of cost-benefit. The benefit was defined as visits by users from a search engine while cost was defined as visits from the web crawlers. Also, the effectiveness of search engines was given a value based on ratio of visit counts by search engine to number of counts for crawlers' visits.

The study by Sun about regulation and behavior of web crawlers was one of the most comprehensive studies that investigated and measured the ethical issues pertaining to web crawlers. Another important study which explored the ethical issues pertaining to web crawlers was by two researchers at the University of Wolverhampton in the U.K., Thelwall and Stuart (2006). Their work examined moral issues in order to build guidelines for web crawlers' creators and owners (Thelwall & Stuart, 2006). Also, the study by Thelwall and Stuart looked into how crawlers can impact privacy, cost, and copyright issues on the web (2006).

d. Web crawler detection and cloaking

The topic of web crawler detection and identification is among one the most important topics that has been investigated by of some of the previously-named researchers, but each study has had its own distinct approach with some challenges in terms of implementation or practical use of the suggested approaches. This group of studies reflected and documented two important fundamental points about web crawler's detection. First, the previous studies pertaining to web crawlers describe some of the well-known challenges with the misuse of web crawlers by hackers and spammers. Second, the previous researchers elaborated and proposed some of the early and basic solutions to this problem of identifying and preventing web crawler along with the limitations of each solution. It is important to emphasize that this study's approach and solution to the problem of identifying and

preventing web crawler are very different from those of previous studies. One of the most important and early studies explained the common weakness of simply identifying the web crawlers by IP address as challenging because using just the IP address as the only identifier of crawlers can be challenging for those web crawlers who hide themselves or replicate someone else's IP address (Tan & Kumar, 2002). This study, which is called *Discovery of Web Robot Sessions Based on Their Navigational Patterns,* investigated the following (Tan & Kumar, 2002):

First, an overview of use of web crawlers was provided, including why there is a need to be able to identify the web crawlers. Among various reasons which the researchers explained, some of e-commerce organizations do not want robots to gather business intelligence on their sites because traffic generated by robots can mislead e-commerce businesses about their customer's visits. Furthermore, the researchers pointed out that robots can also create problems for click-through payments where advertisers pay whenever a user clicks on their ad via website because those clicks may not be a correct reflection of people visiting or seeing an ad.

Second, web robot detection and existing challenges were explained. Some of the important techniques for identifying crawlers were examined, including robot.txt and user agent check.

Third, in terms of approach, the researchers preprocessed the web server logs, which can be very unreliable, and extracted information in order to build a classification model based on label of each session. The researcher also created a metric to evaluate performance.

Fourth, the study used two sets of data to complete this research. Group One was the training data, and it was used to build the models. Group Two included all data sets from user agents. However, the study used only four users' agents to build and test the models.

The research found many previous approaches, such as using robot.txt or just http request alone, are not very accurate indicators of identifying robots; instead, using a navigational pattern is a more reliable and accurate method of detecting crawlers (Tan & Kumar, 2002). The research by Tan and Kumar falls short in the approach because in order to detect web crawlers, four HTTP requests must be submitted to the server, which is too late to detect crawlers at that point. Shortly after Tan and Kumar's study was completed, a few other studies investigated new approaches to address the same problem of identifying web crawlers but in a more effective way. For example, a study called *Characterizing Crawler Behavior from Web Server Access Logs* compared and investigated the crawler's behavior and characteristics versus humans to provide insight about performance, web usage, and design (Dikaiakos, Stassopoulou, & Papageorgiou, 2003). The study examined the following:

First, the study gathered logs from four research organizations which included University of Cyprus, Institute of Computer Science, National Technical University of Athens, and University of Toronto to analyze them and complete the research.

Second, the study found that 33.52 MB http traffic was generated on each server as the result of web crawlers consuming resources. The total http requests for each web server varied, but the minimum was 4.02% and the maximum was 10.32%. The study also found the Get method is used in most of http calls. However, instead of the Get method, the Post method can also be used to submit a request to a web page.

Third, resource referencing or requesting are mostly targeting text and images on the web. The researchers found 90% of all requests were only interested in gathering information in text format and image format instead of other content types such as audio, video, or applets.

However, even by analyzing the web server logs or traffic, it is very difficult to accurately and consistently detect web crawlers all the time. So, a new approach was needed to identify humans versus robots or web crawlers. The new solution was to create a test which humans can pass but computer would fail, also known as CAPTCHA, short for Completely Automated Public Turing Test to Tell Computers and Humans Apart (Von, Blum, & Langford, 2004). For example, a distorted image would be presented; humans could easily identify what the image is, but computers would fail to process it. The study by Von, Blum, and Langford pointed out that there will be a day when computers will be able to pass current tests such as distorted images. So the more recent studies attempted to create a way to identify web crawlers that would not require web users to take a test for logging into a website or visiting a web page. One of the relatively recent studies which proposed a new path to solve the crawler detection problem examined clickstreams of machines versus humans (Lourenco & Belo, 2006). *Clickstream* is tracking of screen or links which users click on (Wang & Lee, 2011). In addition to the crawler identification, a new challenge about cloaking has been documented by researchers (Wang, Savage, & Voelker, 2011; Wu & Davison, 2006). Cloaking is the technique that presents different contents to humans and crawlers (Lin, 2009). Cloaking is not acceptable anymore for most search engines because search engines fail to return correct results when cloaking is implemented on web pages (Lin, 2009).

e. Deep Web and Web Crawlers

Most current web crawlers are very efficient at indexing static web pages but not very efficient at indexing dynamic web pages. The dynamic web pages are those web pages which are only viewable after a query is submitted to a server (Artail & Fawaz, 2008). A web server always generates content and sends the results back to the client after receiving an http request (Artail & Fawaz, 2008). The static pages are the opposite of dynamic web pages because their content does not change and the web pages do not require a web server processing a query to generate a web page. It is very difficult to use crawlers for dynamic web pages because crawlers need to write a query and then process the results, which requires a complex algorithm and uses a lot of processing resources. Various researchers have attempted to investigate the crawling of dynamic web pages in order to solve the problem of indexing dynamic web pages. Most often the term *deep web* is used to refer to the information on the web which is very difficult to reach by most common web crawlers because the content of pages is created dynamically (Ke, Deng, Ng, & Lee, 2006). One well-known study about deep web and crawling mechanisms proposed writing a query and submitting it to web servers by selecting random keywords, generic frequency keywords, or adoptive keywords (Ntoulas, Zerfos, & Cho, 2005). Selecting random keywords is not very efficient because it can use a lot of web resources to return useable results (Cafarella, Halevy, & Madhavan, 2011). Generic frequency works by using generic document corpus collected elsewhere (say, from the Web) and obtaining the generic frequency distribution of each keyword, which is still not very practical because it can be difficult to consistently write correct queries given the nature of web pages and changing content of web (Ntoulas, Zerfos, & Cho, 2005). Adaptive keyword selection is simply using previous submitted queries and

analyzing those results in order to select a correct keyword for creating a new query (Ntoulas, Zerfos, & Cho, 2005). The solutions presented by various researchers to address the deep web are still not very practical; for example, one study created more than 13,000 queries to send a request to web page (Madhavan, Ko, Kot, Ganapathy, Rasmussen, & Halevy, 2008). Another approach recommended by one of the most recent studies involved using focused crawling for a given domain (Sharma, & Sharma, 2011). The researchers used online book websites along with focused crawling to write a more targeted query with more accurate results (Sharma & Sharma, 2011).

f. Miscellaneous study related to crawler

Last, some of the studies pertaining to crawlers were very distinct and did not fit into the previously mentioned topics. For example, some studies focused on reconstructing web pages using crawlers when the backup copy of a web page is not available (McCown & Nelson, 2006). This study proposed a process by which the information on the web page could be retrieved from Google-, Yahoo-, and MSN-cached information. Another study focused on security and using web crawlers as a resource to identify malicious software on the web (Likarish, & Jung, 2009). Also, one study suggested utilizing web crawlers for building a digital library (Pant, G., Tsioutsiouliklis, Johnson, & Giles, 2004).

Summary

This chapter presented information about the background of web crawlers and a literature review pertaining to this study. The background section provided information about web crawlers and some of the challenges such as big data processing and how web crawlers are needed to process massive amounts of data on the Internet and social platforms. Also the stakeholders of web crawlers were described, including how they use or misuse web crawlers

for their benefit. The literature review section focused on web crawlers and significant studies in terms of approaches and solutions related to web crawlers. The main points of each study were described in detail about Robots Exclusion Protocol, web caching and performance optimization, ethical aspects of crawlers, web crawler detection and cloaking, deep web and web crawlers, and miscellaneous studies related to crawler topics.

Chapter 3. Methods

Introduction

This chapter will provide details about the design of this quantitative study, data collection methods, and the sample population. The focus of this study was to identify a novel defense mechanism against web crawler intrusion without cloaking. This study utilized a quasi-experimental design to investigate whether the five-factor identification process can prevent web crawlers from visiting web pages. This chapter is one of the most important parts of this study because it explains all the steps it takes to accurately prepare and implement a quasi-experiment and measure the results.

Research Design

One of the main objectives of this research was to choose the best research design to allow a comprehensive and effective investigation and analysis of web crawlers and web pages. In addition, this study investigated cause and effect of the novel five-factor identification given the limitations and available resources. Since this study concentrated on investigating the cause and effect, an experimental design was selected as one of the best approaches of research design to conduct this research. According to a book called *Practical Research: Planning and Design,* "A researcher can most convincingly identify cause-and-effect relationships by using experimental design" (Leedy & Ormrod, 2005). However, there are various types of experimental designs that were also considered to investigate the cause and effect relationship. For example, pre-experimental designs can be used for studies where it is very challenging to study the cause and effect because the independent variables do not change, control groups do not have randomly selected entities, or the control groups are very

similar (Leedy & Ormrod, 2005). Since this study had multiple control groups and the groups had the same size in pre- and post-test groups, this approach was not selected. Also, using only pre-experimental designs is more challenging to determine the cause and effect systematically since there are other approaches. A more comprehensive approach can be used such as true experimental design or quasi-experimental to address some of the weaknesses of pre-experimental design, such as failing to make sure the control groups are similar by comparing them prior to or after conducting the study. A true experimental design provides much greater control and better results with higher internal validity because the sample population is selected randomly (Leedy & Ormrod, 2005). One of the most important aspects of true experimental design is the ability to select random samples from a population, but this is not always possible. In cases or studies where a true experimental research design is not possible to implement, a quasi-experimental design might be an alternative approach to investigate a cause and effect relationship (Pew & Hemel, 2004). Therefore, this study used nonrandomized control groups pretest-posttest because this research conducted an experiment to see whether the novel five-factor identification which uses pass key, date, user agent, IP, number of visits for the web server/page (allowed each day) can truly prevent web crawlers from entering web pages or servers by using nonrandom samples. So there are two main reasons for selecting and using the nonrandomized control groups pretest and posttest. First, a true experimental design was impossible because of practical challenges involved in acquiring hundreds or thousands of domains names and servers to test the hypotheses. Other studies have also recommended using quasi-experimental designs where “randomization may not be viable due to economic and experimental integrity concerns” (Oktay, Taylor, & Jensen, 2010). The second reason for selecting a quasi-experimental method was the clear

advantage of being able to control the entire classes of variables over other methods for experimental research. According to Jensen, Fast, Taylor, and Maier, "QEDs can surpass the validity of attempts at statistical control because they can control for entire classes of variables, even though those variables are not identified, measured, or modeled" in a randomized way (2008).

The nonrandomized control group pretest-posttest is best described as an approach between the static group comparison, which is a pre-experimental design type, and pretest-posttest control group design. It is even documented that a nonrandomized control group has a clear advantage over a randomized control group in some cases because it involves two groups that are not randomly selected in the same way as static group comparison, but it uses pretreatment observation in same way as the pretest-posttest control group design of true experimental design (Leedy & Ormrod, 2005). There were three main steps which were completed as part of the experiment in addition to data collection and analysis. The first step was to conduct this experiment for the pretest step. In order to complete the pretest and posttest, a Java application was created along with 90 web pages for each control group on two separate computers and web servers. Furthermore, a web crawler program was created to gather and download web pages. The pretest for this study used a Java web program on the server side to render and create web pages and a separate web crawler which would crawl to two groups with each group having 90 web pages. Whenever a crawler visited a web page, it attempted to download the web page onto a local computer. Once the pretest was completed and results were analyzed and stored in the database, a treatment was introduced. This treatment introduced the novel five-factor identification process which used pass key, date, user agent, IP, and number of visits for the web server/page (allowed each day). Once the

treatment was introduced, a posttest was conducted. The posttest steps were very similar to pretest steps except this time, for a web page to render its contents, the five-factor identification was presented by the web crawler first, and if those values matched with web servers' values for each key, then the web crawler was allowed to download the web page; otherwise a blank web page with a warning message was visible to the crawler only. Once all of the above steps were completed, the results were analyzed using SPSS.

Measurements

For the pretest and posttest, the following *s* dependent variable was measured where *s* was the crawler's number of visit to each web page. The value for *s* was calculated by counting the number of downloads by crawlers. This approach, which counted the number of downloads by crawler, was selected because previous studies have also used this technique when measuring crawlers' success or failure (Kumar & Vig, 2009; Radhakishan, Farook & Selvakumar, 2010). For example, if a crawler was not able to download a page, then the value for *s* was set to zero because the web page was not downloaded and this was considered a success since the goal of this study was to find a new way to prevent web crawlers from downloading web pages without cloaking.

The dependent variable studied for this research was the following:

- *S* success or failure visits for web crawler which attempted to download. Zero indicates success (because the web page was not downloaded) and one indicates failure (because the web page was downloaded by web crawler).

The independent variables were *u, i, t, p,* and v, which are defined below, where1 represents success and 0 indicates failure.

- *u* is the success or fail match of user agent for crawler vs. web page.

- t is the success or fail match of time for a crawlers visit.
- *p* is the success or fail match of passkey for crawler vs. web page.
- *i* is the success or fail match of IP addresses valid to visit a web page.
- *v* is the success or fail match when number of visits (allowed each day) matches what the server expects.

Research Setting

The research setting was 10 computers with multiple web servers and 72 web pages on each computer for each webserver. In addition, a web crawler was hosted on dedicated web servers which had access to reach to all computers and the dedicated web page servers with 90 web pages on the local area network for each group. As depicted in Table 3, a total of 720 web pages were used for this study.

The experiment consisted of testing for two main types of crawlers by creating 90 web pages for each group to test for accessing each web page. The first group as depicted in Table 3 was constructed to test and make sure the treatment did not inadvertently prevent valid web crawlers to visit web pages. One example of valid web crawlers being impacted by invalid web crawler prevention mechanisms is where search engine web crawlers such as googlebot might be prevented from indexing web pages even though bad/invalid web crawlers might have been effectively prevented by a web crawler prevention mechanism. The second group as depicted in Table 4 was constructed to test for effectiveness of treatment for preventing unwanted crawlers. This test replicated a process where a hacker may use a crawler to download a web page. The test steps involved completing pretest and posttest steps consistently for both groups. The pretest step involved testing and storing a number for web crawler success or failure visit for each web page in Groups 1 and 2 in Table 3 for both

valid and unwanted groups. After completing the pretest step, a treatment was introduced to Group 2 only for both valid and unwanted groups. The treatment consisted of introducing an agent on the web pages for existence of valid matches of independent variable values (please see Measurement section for variables list and explanation). For example, if a crawler was able to visit a web page and all independent variables matched, then an *s* value was set at 1.

Table 3

Valid Web Crawlers

Web pages with valid web crawlers			
Pretest		Posttest	
Group 1:	90 web pages	Group 1:	90 web pages
Group 2:	90 web pages	Group 2:	90 web pages
Treatment for group 1 : an agent on the web pages in group 1 to check for existence of valid matches of u, t, p ,I and v(please see Independent variables list and explanation in the following page)			

Table 4

Unwanted Web Crawlers

Web pages with <u>unwanted</u> web crawlers			
Pretest		Posttest	
Group 1:	90 web pages	Group 1 :	90 web pages
Group 2 :	90 web pages	Group 2 :	90 web pages
Treatment are only for group 2: an agent on the web pages in group 2 to check for existence of valid matches of u, t, p ,i and v (please see Independent variables list and explanation in the following page)			

Population, Sample, and Subjects

In order to select an appropriate sample size, first we must learn about the size of the population. However, this study is investigating web pages, and it is impossible to know exactly how many web pages are on the web (Westfall, 2009). Also, previous researchers have selected a limited number of web pages to study even though there were many web pages available on the Internet (Dalvi, Machanavajjhala, & Pang, 2012). For example, one study used only nine different domains to study structured data on the web (Dalvi, Machanavajjhala, & Pang, 2012). Furthermore, the challenge of selecting the correct number of web pages has been documented by other researchers, and one of the methods suggested is convenience sampling (Wang, 2006; Blank, Fielding, & Lee, 2008). So this study uses the similar approach to a study called *An Analysis of Structured Data on the Web* (Dalvi, Machanavajjhala, & Pang, 2012). In Figure 2, multiple web servers are used, but the sample size was 90 web pages for each group. The groups are defined in Tables 3 and 4.

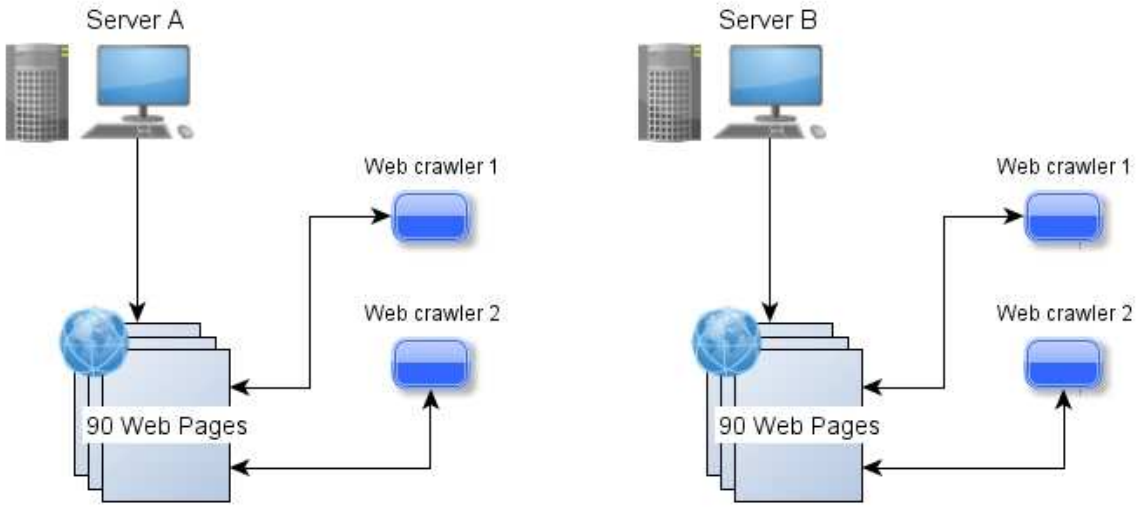
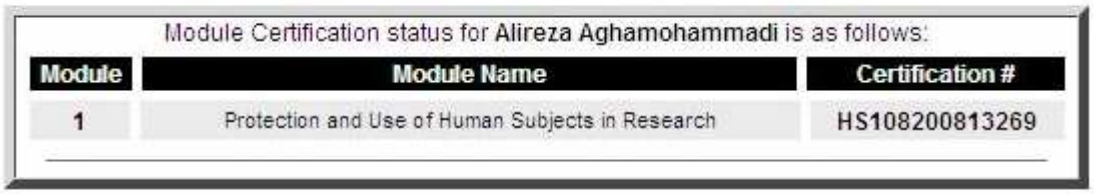


Figure 2. Servers and Web pages

Humans Subjects Approval

This study did not involve any human subjects for testing or collecting data. Only the researcher was conducting the experiment and the process only involved software applications and data. However, online training modules were completed from UHSRC at EMU.



Module Certification status for Alireza Aghamohammadi is as follows:

Module	Module Name	Certification #
1	Protection and Use of Human Subjects in Research	HS108200813269

Data Collection

Data collection was critical for this study because the process needed to be done systematically and accurately. In order to make sure the process was done this way, computer programs were used to automate and collect data for all groups in pretest and posttest steps. In order to write web crawlers and create web pages, various programming tools and software were used. First, an Eclipse tool was used to write the web crawlers and create web pages in JSP (Java Server Pages). Eclipse is IDE which is an Integrated Development Environment for building applications. Second, Tomcat web servers were used to host the web pages. The language selected for programming the web crawlers and collecting data was Java. Java application was the main instrument used to collect data and track the success or failure of download or uploads.

Two types of web crawlers were used for this study: an unwanted web crawler a valid web crawler. These two web crawlers attempted to download 90 web pages into a local folder. In terms of approach for collecting data, this study followed similar steps as previous studies (Chen, Bhowmick, & Nejdl, 2009). The following are the steps which were completed to gather and collect data for pretest:

- First, web crawlers were created on a web crawling host server, and a folder on the host was created to collect and gather information about the crawling. For example, if a web crawler wanted to download page 1, first it created a file under a folder which was called:

  C:\phd_data\wanted\group_X\pre_test

  The X was replaced by the number of group type, 1 or 2. There were two web crawlers as indicated previously: an unwanted one and a valid web crawler.
- Second, the web crawler submitted a request to the web server to download a web page.
- Third, the web crawler stored the downloaded web page into a file in an html format with the following naming convention in the directory defined in step one.

  IP address _ Port number _ date_ time stamp _ output.html
- Fourth, results were processed and stored in a database for further processing in the data analysis step.

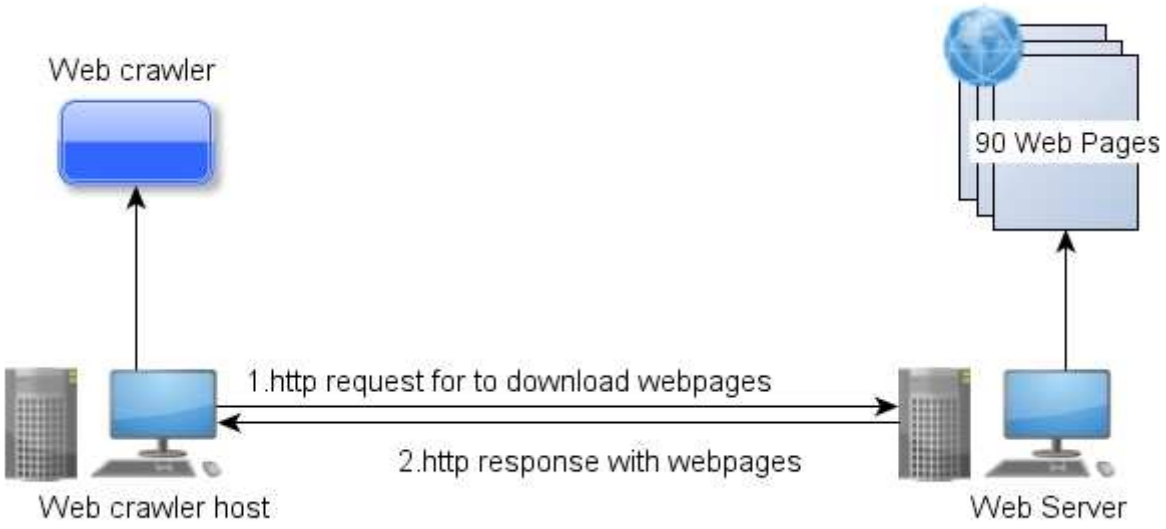


Figure 3. Request response process

The above steps were done twice because there were two web crawlers for each group.

After completing the pretest steps, the results were examined to make sure there were no duplicated IP addresses and that the time on the servers did reflect the actual time when the experiment was conducted. The downloaded pages were also examined to make sure they were not blank.

The treatment introduced in the posttest step was the novel five-factor identification process which used pass key, date, user agent, IP, and number of visits for the web server/page (allowed each day). This treatment consisted of changing web pages to check for above values before rendering the content of web pages. The steps for collecting posttest data were:

- In order to make the pages download, first directories were created. In addition to creating the directories, the web crawler program was also constructed although the web crawler functionality was very similar to the pretest step and essentially it was

the same web crawler. A folder created on the web consisted of the following format:

C:\phd_data\unwanted\group_X\post_test

The X was replaced by the number of group type, 1 or 2. There were two web crawlers as indicated previously; one was unwanted and the other was a valid web crawler.

- After creating the directories for the posttest crawling step, the web crawler submitted a request to the web server to download each web page.
- In addition to crawling, the results were downloaded into an html file, and the file name had the following naming convention in the directory, defined in step one, to be able to identify each web page individually and distinctly.

  IP address _ Port number _ date_ time stamp _ output.html
- Fourth, results were processed and stored in a database for further processing in data analysis step similar to the pretest step.

At the end of completing the posttest steps, the stored IP values were examined. In addition, the key values were stored in the database to be sure the values were not null or blank. Also, the counts of total web pages were compared against the database to make sure they both downloaded and stored 720 web pages each.

Data Analysis

After data collection, the data analysis was completed. The data analysis is an important aspect of any research because it is a process of analyzing data systematically and logically to describe, summarize, and evaluate data. In this study, the Binary Logistic Regression Analysis, also known as Binary LR analysis, was selected to analyze the data. Logistic regression provides a mechanism to analyze a dichotomous response variable where

outputted data or the dependent variable is in a binary format (Bewick, Cheek, & Ball, 2005). All the data analysis and binary logistic regression were done using IBM SPSS software. There were three main reasons for choosing logistic regression for this study, and the following paragraphs describe these reasons in more detail as it pertains to this study.

First, previous studies have used this approach and it is a proven mechanism given the goal and limitations of this study (Salem, 2001; Qureshi, 2006). Second, since the data and measurements are dichotomous (binary format), other methods such as analysis of variance (ANOVA) would not be a good approach instead another method such as logistic regression is more suitable for this study because "logistic function f (z) ranges between 0 and 1" and it is simple and popular to use in various studies (Kleinbaum, & Klein, 2010). Third, logistic regression analysis will confirm or refute the treatment effectiveness in relation to the outcome or independent variable in terms of probability. There are two main groups under this study—the treatment/intervention group and the control group—and no intervention is exposed to this second group. The main function of data analysis was to compare these two groups by calculating the *p* value. Observed significance level, or p-value, "is the probability (assuming H_0 is true) of observing a value of the test statistic that is at least as contradictory to the null hypothesis and supportive of the alternative hypothesis, as the actual one computed from the sample data" (McClave, Benson, & Sincich, 2001). Also, the value of alpha (α) indicates that the significance level of the test was set at .05 and the confidence interval was at 95%. When the significance level is set at .05, it means the finding of a study only has a five percent chance of not being true and a 95% chance of being true.

In linear regression $Y= \beta_0 + \beta_1 X + \varepsilon$, Y indicates the result or the dependent variable. X is the independent variable, β_0 is the intercept, β_1 is the slope, and ε is the Model errors

(Ramirez & Ramirez, 2009). Unlike linear regression, multiple regression has multiple independent variables, but the general function is very similar to linear regression (McClave, Benson, & Sincich, 2001). The logistic regression is different from multiple/linear regression because it deals with predicting the probability of Y value, but it is similar to multiple/linear regression in a way because the general function "aim is to write the conditional expectation of the dependent variable Y as a linear combination X" in terms of regressing (Tuffery, 2011). The binary logistic regression is very similar to logistic regression, but it deals with only one binary dependent outcome of Y = 0 or 1, and below is the equation of logistic regression (Sheather, 2009):

$$P(Y) = \frac{1}{1 + \exp(-\{\beta_0 + \beta_1 X\})}$$

Where $\beta_0 + \beta_1 X$ will be calculated as the result of solving the equation, P is the probability and exp is exponential function and Y is the dependent variable. The calculation for the binary logistic regression will be completed in SPSS. The following variables in the equation, classification, and cross tabulation tables will be produced, and the results will be used to confirm or refute the hypotheses.

The classification table generated by logistic regressions process and SPSS was created to better understand the web crawler's success or failure. In the classification table, zero indicates success (because the web page was not downloaded) and one indicates failure (because web page was downloaded by web crawler). Table 5 shows a sample classification table.

Table 5

Classification Table Example

Classification Table

Observed			Predicted		
			success or Failure visits for valid web crawler which attempted to download		Percentage Correct
			.00	1.00	
Step 1	success or Failure visits for valid web crawler which attempted to download	.00	##	##	%##
		1.00	##	##	%##
	Overall Percentage				%##

For example, if the overall percentage is 80, then it indicates that 80 percent of web pages were not downloaded by web crawlers. In addition to the classification table, the variables in the equation table will be generated as part of the binary logistic data analysis process. Table 6 is a sample of variables in the equation table:

Table 6

Variables in the Equation Example

Variables in the Equation

		B	S.E.	Wald	df	Sig.	Exp(B)
Step 0	Constant	1.386	.791	3.075	1	.080	4.000

The most important aspect of the Table 6 information is the Sig. which indicated the *p* value.

Analysis Tools

The IBM Statistical Product and Service Solutions (SPSS) version 18 was used for this research, and a comprehensive data analysis using binary logistic regression was

completed for the results of pretest and posttest, and treatment groups and control groups were compared. In addition to logistic regression and various tables such as a pie chart, tables about the number of pages downloaded and how each key performed for each group were created.

Validation

Validity is in an important aspect of research because it addresses the "accuracy, meaningfulness and credibility of the research as a whole" (Leedy & Ormrod, 2005). This study focuses on validity by making sure the conclusion and measures are acutely reflective of the collected data in a meaningful way. Following are the list of validity threats along with an explanation about how this research has attempted to address those validities (Isaac & Michael, 1981; Campbell & Stanley, 1973):

Face validity: In order to make sure this study's experiment and test has face validity, multiple previous studies were examined to confirm the instrument and measurements were similar. The previous studies used Java and session counting as a way to measure and test web crawlers' visits (Lourenco & Belo, 2006; Fraternali, 1999).

History: There are no specific events that could occur to impact the participant or the measurement between the pretest and posttest except malware or a virus. So in order to make sure malware or a virus do not impact the study, anti-malware software and anti-virus software will be used as a precaution prior to and after pretest and posttest.

Maturation: This study only uses webserver, web pages and programs for a short period of time. So the subjects of this study will not change over time because the subjects of this study are web pages and web crawlers. Furthermore, the webserver and web crawlers

will not be running for days or months for each experiment. So the performance, measurements, and results should not be impacted.

Testing: Some studies may not have correct results or measurements because the pretest process impacts the posttest process. This study uses two different servers, and on each server 90 web pages will be tested for each group in isolation to alleviate the pretest impacting the posttest. The study does not plan to run the experiments in parallel.

Instrumentation: Changes in instrument, observers, and so on can sometimes create different results. However, the instrument for this study is consistent for all the tests and groups, and it does not change because only Java application, which uses sessions for measuring the crawlers, is used.

Statistical Regression: Some studies may accidently select subjects or individuals because of having extreme scores or performance. This can impact the results because the posttest results might show a great improvement because the lowest score or subject was selected. This research uses web pages which are only replications of average web pages; the web pages used and studied for this research are replications of typical web pages with some html code text, images, videos, input box, select box, and table. So the web pages used are not too content heavy with various multi-media components, such as video clips on you tube, and are not as simple as text web pages.

Selection: Selecting subjects for the study is very important because if the selected groups are not equal, then the results will be impacted. In other words, the data analysis, hypothesis testing, and conclusion will be done based on wrong information. Since nonrandomized control group pretest-posttest design is selected for this study, the groups are

controlled, and the web pages used for this study are same number in each group for the pretest and posttest experiment.

Selection-Maturation Interaction: In some studies the selection may interact with maturation, meaning that one can impact the others. For example, two groups, old and young, might be trained on a tool but when tested, a young person may perform better or worse as a result of their age or experience. This research will be conducted by using web pages with similar characteristics and will be done during the same time period but not exactly at the same time. So selection-maturation interaction will not influence this study.

Mortality: In some studies, the subject loses interest or does not want to participate any longer, and that would be a concern with completing research. This study used web pages and Java programs, and mortality does not really apply for this investigation. However, a backup of all the programs and web pages are created for traceability purposes.

Personnel

Only the researcher was responsible for collecting the data; however, an assistant and recommendations from following committee members were needed:

- Dr. Ali Eydgahi, Ph.D., (Chair)
- Dr. Daniel Fields, Ph.D.,
- Dr. Huei Lee, Ph.D.,
- Dr. Alphonso Bellamy, Ph.D.

Budget

The cost for this research was very low since there was no need to purchase data from a vendor or organization, but a flash drive, new computer, SPSS software were needed as depicted in the chart. The most expensive items for this research were computer and IBM

SPSS software. However, in order to even further minimize the cost for this research, SPSS software from library computer labs were used, but the estimated cost of SPSS is also provided in the chart for the future researcher to better estimate and plan the cost of similar study.

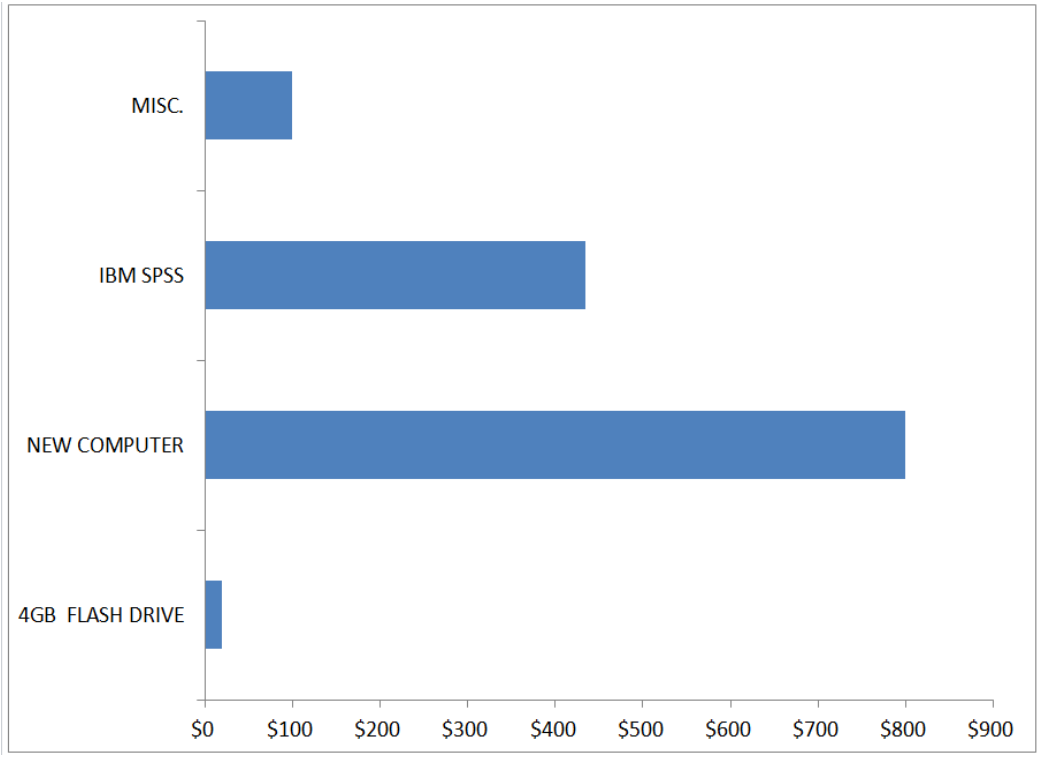


Figure 4. Budget

Timeline

The timeline in Table 9 was proposed to complete this study. The duration and time proposed for each task was an approximation of the expected amount of time it would take to complete each task.

Table 9

Timeline

Task #	**Tasks**	**Start date - Duration**
1	Downloading the software and setup	5/20/2013 - one week
2	Execute experiment and analyze	6/1/2013 - four weeks
3	Completed the statistical steps	7/1/2013 - four weeks
4	Compile and review	8/19/2013 - two weeks
5	Organize and prepare last copy after review	9/1/2013 - three weeks

Summary

This chapter provided information about research design and why the quasi-experiment is selected for this study. The measures and research settings were also documented and explained. The population, sample, and subjects were presented, along with the justification about the approach taken for this study in terms of selecting samples. This chapter also explained the human subject approval process along with data collection, data analysis, personnel, budget, and timeline.

Chapter 4. Results

Introduction

This chapter presents information about the results of collected data and data analysis. First, summary information about the web crawlers' return rate as it pertains to the results of all collected data and five-factor identification pretest and posttest are provided. Second, demographic characteristics of the sample are presented along with the results of pretest and posttest analysis. Third, research questions/hypotheses results are presented with information about how each hypothesis was rejected or not rejected based on statistical analysis outcome. Also, this chapter categorized the results into two main groups, as was explained in Chapter 3. The first group targeted the unwanted web crawlers, and the second group included the wanted web crawlers. However, each group then was subdivided into pretest and posttest subgroups, and the results are presented based on the pretest and posttest. SPSS and Binary Logistic Regression were used to create the results in this chapter because of the nature of dichotomous data and accurate processing of data. Also, data reliability information, including data validity results and Cronbach's alpha information, are provided and explained in this chapter. Last, this chapter provides relevant charts and tables, but more detailed information about the SPSS outputs are in the Appendix A and B sections.

Web Crawler's Return Rate

This study used Java software/application to collect data for this research, and the results were gathered and documented. There was no survey used for collecting data steps; instead, multiple web crawlers were used to download web pages. There were a total of 720 web pages on 10 computers and webservers as depicted in Table 10. Web crawlers were hosted on a single server but attempted to go to multiple computers, while each Apache web

server hosted web pages on a local area network. The web crawlers were categorized as wanted and unwanted prior to pretest and posted. A return rate is typically used for studies with surveys, and it is calculated based on the number of completed samples divided by the total sample size (Basarab, 2010). This study did not use any surveys, but the web crawlers' download can be viewed in context of attempted download of web pages. Following are the information that was gathered for calculating a return rate. Among 720 web pages, only eight did not download as result of error 404 (or web page not available). So 712 web pages were crawled without any web page errors; indicating a very good expected return rate. Therefore, the return rate for pretest and posttest was 98% since the total sample size was 720 and crawled webpages with no error was 712.

Table 10

Web page counts per server

Web Page Counts	Host /Server
72	http://192.168.0.114:8080/
72	http://192.168.0.107:8080/
72	http://192.168.0.113:8080/
72	http://192.168.0.100:8080/
72	http://192.168.0.119:8080/
72	http://192.168.0.106:8080/
72	http://192.168.0.126:8080/
72	http://192.168.0.111:8080/
72	http://192.168.0.128:8080/
72	http://192.168.0.110:8080/

Demographic Characteristics of the Sample

The sample size used for this study was 720 web pages. The web pages were crawled by using two types of web crawlers, a wanted/good web crawler and an unwanted/bad web crawler. There were 90 web pages per each group (9 web pages crawled per computer). The total computers used for this study were 10, excluding a computer for hosting web crawler application. The test types were categorized to pretest and posttest for each web page, and two web crawlers were used to visit the web pages as depicted in Table 11. The groups were categorized to Group 1, indicating treatment was not introduced, and Group 2, indicating that the five-factor identification/treatment was introduced only to posttest step. Table 11 contains the total number of web pages which web crawlers attempted to download by test type, web crawler type, and group type.

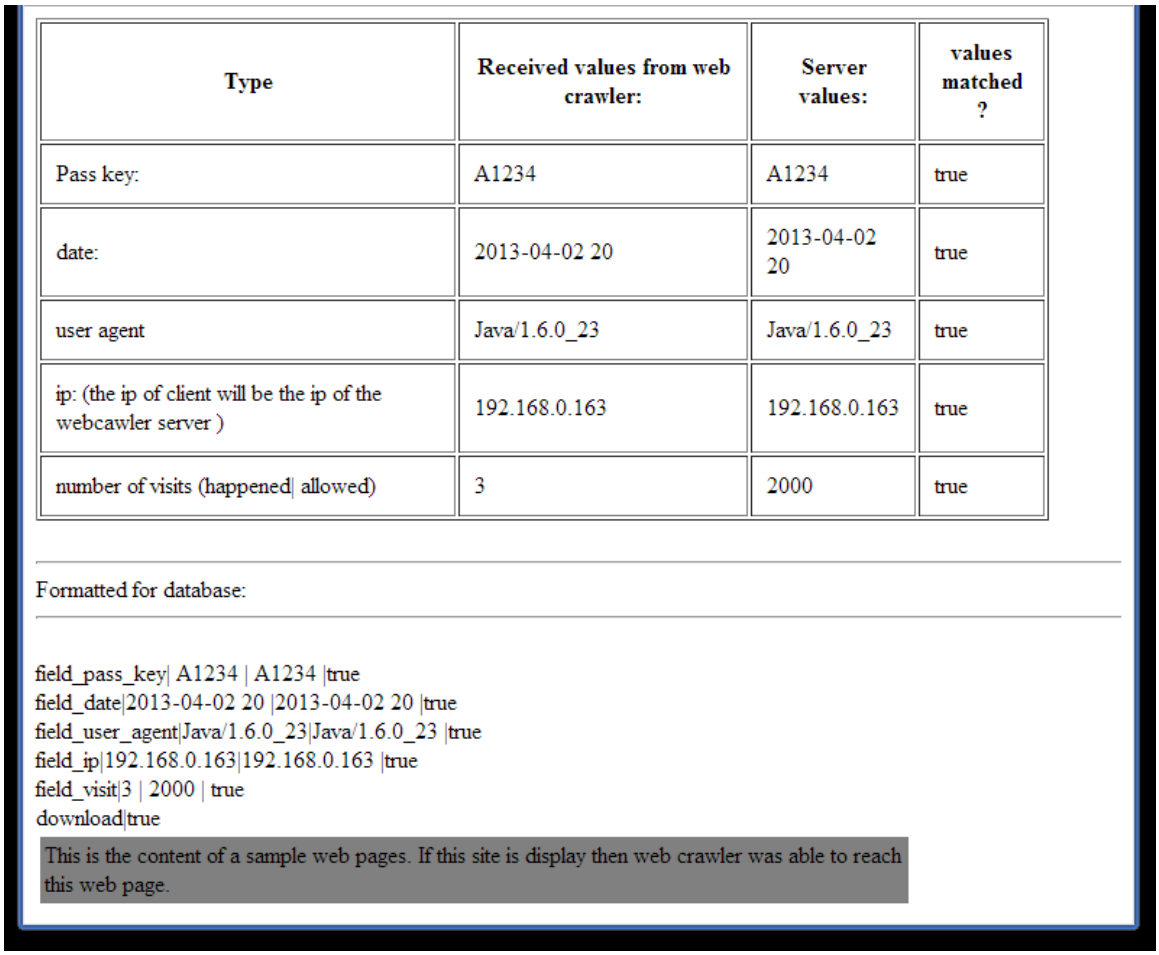
Table 11

Sample Demographic

Web Page Count	Test Type	Web Crawler Type	Group Type
90	pretest	unwanted	group_1
90	posttest	unwanted	group_1
90	pretest	unwanted	group_2
90	posttest	unwanted	group_2
90	pretest	Wanted	group_1
90	posttest	Wanted	group_1
90	pretest	Wanted	group_2
90	posttest	Wanted	group_2

The collected sample data involved using two web crawlers to download each web page, and results were loaded into a database along with a download and formatted HTML file. The main reasons for storing the results in two locations were validity, reliability, and

traceability. If a web crawler was able to download a web page, then the message “This is the content of a sample web pages. If this site is displayed then web crawler was able to reach this web page” was displayed, as depicted in Figure 5.



Type	Received values from web crawler:	Server values:	values matched ?
Pass key:	A1234	A1234	true
date:	2013-04-02 20	2013-04-02 20	true
user agent	Java/1.6.0_23	Java/1.6.0_23	true
ip: (the ip of client will be the ip of the webcawler server)	192.168.0.163	192.168.0.163	true
number of visits (happened| allowed)	3	2000	true

Formatted for database:

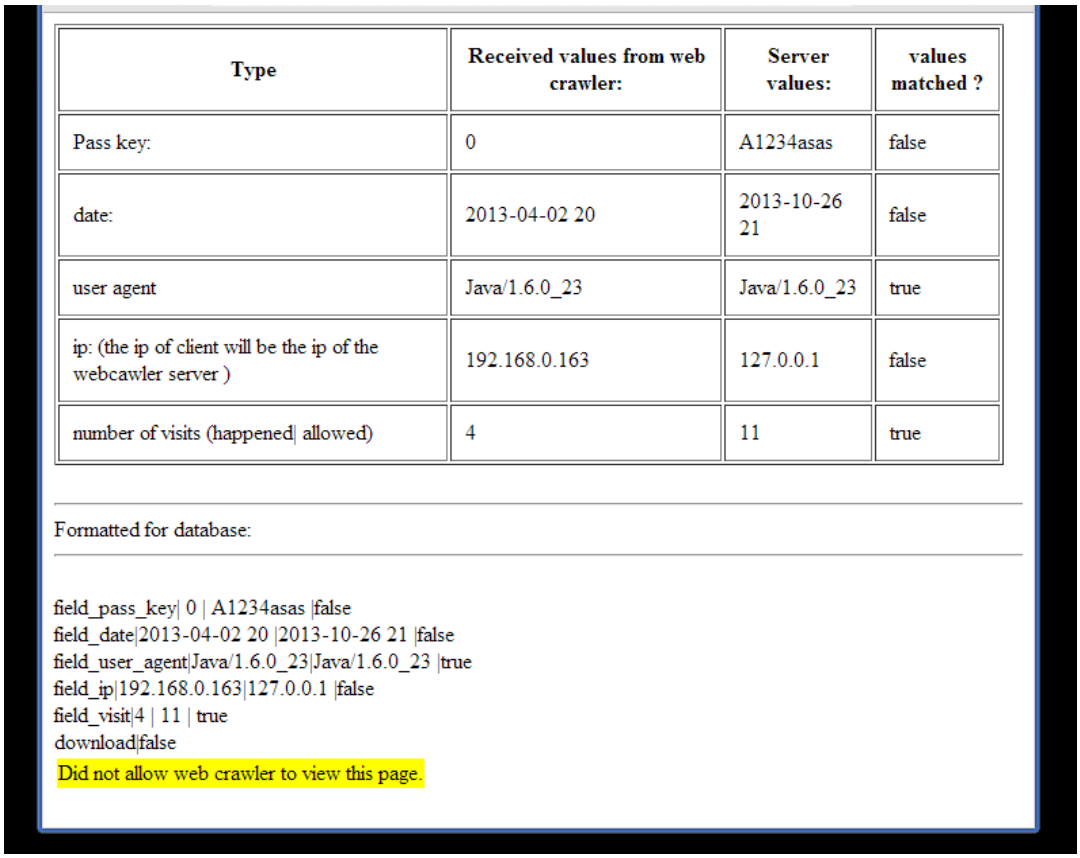
field_pass_key| A1234 | A1234 |true
field_date|2013-04-02 20 |2013-04-02 20 |true
field_user_agent|Java/1.6.0_23|Java/1.6.0_23 |true
field_ip|192.168.0.163|192.168.0.163 |true
field_visit|3 | 2000 | true
download|true

This is the content of a sample web pages. If this site is display then web crawler was able to reach this web page.

Figure 5. Sample web page when web crawler was able to download.

On each web page the pass key, date, user agent, IP, and number of visits for the web server/page and web crawlers were displayed and collected (if the values were available or sent to web server). Also, a result table was displayed where five-factor identification keys were used (see Figure 6). If the values were not presented for pretest process, then the values were set to false, indicating that a page was displayed by none of five-factor identifications.

In addition to five-factor identification values, formatted results were displayed and saved too under the main table (see Figures 5 and 6). The formatted values were required to be able to process and store the results in the database for data analysis steps. On the other hand, if a web crawler was prevented from visiting a page, then a message “did not allow web crawler to view this page” was displayed as depicted in Figure 6 (in addition to the five-factor identification and formatted values for database).



Type	Received values from web crawler:	Server values:	values matched ?
Pass key:	0	A1234asas	false
date:	2013-04-02 20	2013-10-26 21	false
user agent	Java/1.6.0_23	Java/1.6.0_23	true
ip: (the ip of client will be the ip of the webcawler server)	192.168.0.163	127.0.0.1	false
number of visits (happened| allowed)	4	11	true

Formatted for database:

field_pass_key| 0 | A1234asas |false
field_date|2013-04-02 20 |2013-10-26 21 |false
field_user_agent|Java/1.6.0_23|Java/1.6.0_23 |true
field_ip|192.168.0.163|127.0.0.1 |false
field_visit|4 | 11 | true
download|false
Did not allow web crawler to view this page.

Figure 6. Sample web page when web crawler was prevented to download.

Web crawlers were able to crawl to 720 web pages, but only 623 web pages were downloaded. It is important to mention that among 97 web pages that web crawlers could not download, only 8 were due to the web page not being available on the network. However, those web pages were not excluded from the data analysis steps and results because on the

Internet and larger networks, a similar outcome is expected. So from total of 97 web pages, 89 web pages were prevented by web server status in addition to those 8 web pages which did not download due to not being available on the server. In other words, 89 web pages were not downloaded by web crawlers because the web page's identity did not match the access keys, and download permission was denied. However, in terms of overall percentage of total web pages for this study, 13.47% were prevented from downloading. The majority of web pages were downloaded, but some were prevented because the five-factorial identification process prevented unwanted web crawlers from downloading web pages. On the other hand, 86.53% of total web pages were downloaded as depicted in Figure 7.

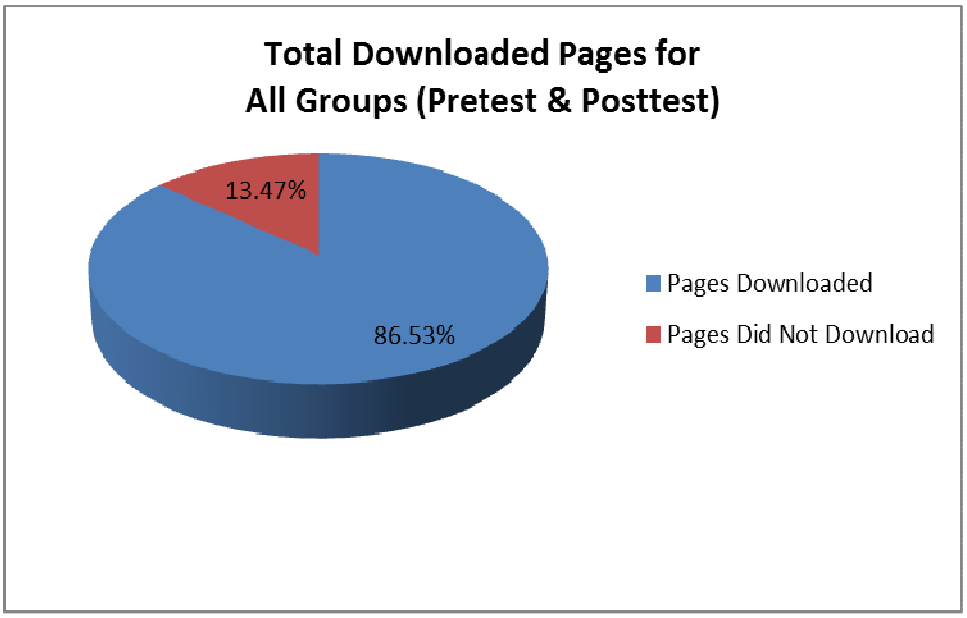


Figure 7. Downloaded Pages

Notice the chart provided in Figure 7 includes all web pages, pretest groups, and posttest groups for wanted and unwanted web crawler types. So in order to have more detailed and deeper results, the following sections will provide four separate groups for which results are documented and explained in a more comprehensive way. The first group is

about characteristics and results of unwanted web crawlers results for Group 1. The second group will provide information about the results of unwanted web crawlers results for Group 2. The third group will provide information about wanted web crawlers results for Group 1 and the fourth group will elaborate about wanted web crawlers results for Group 2.

Unwanted Web Crawlers Results Group 1 (pretest-posttest control group)

One of the main distinct characteristics of this group was the lack of exposure to treatment, and only unwanted web crawlers attempted to download the web pages. The unwanted web crawler pages were downloaded by a web crawler for Group 1 web pages. Group 1 was not exposed to five-factor identification because this was the control group. Among 90 web pages for each group, the unwanted web crawler group was able to access 89 web pages successfully, but one web page in the pretest group and one in the posttest group were not downloaded due to the page not being available. The pretest and posttest results were very consistent; this was expected because the web pages had no mechanism to prevent the pages from accessing and downloading by web crawler. The results are provided in Tables 12 and 13.

Table 12

Unwanted Web Crawlers Results, Group 1

Count	Test Type	Crawler Type	Group Type	Downloaded
1	pretest	unwanted	group_1	FALSE
89	pretest	unwanted	group_1	TRUE
1	posttest	unwanted	group_1	FALSE
89	posttest	unwanted	group_1	TRUE

Table 13

Validation of Five Factorial Keys for Unwanted Web Crawlers, Group 1

COUNT	IP CHECK	PASSKEY CHECK	VISITED CHECK	DATE CHECK	AGENT CHECK	TEST TYPE
89	FALSE	FALSE	FALSE	FALSE	FALSE	pretest
1	FALSE	FALSE	FALSE	FALSE	FALSE	pretest
89	FALSE	FALSE	FALSE	FALSE	FALSE	posttest
1	FALSE	FALSE	FALSE	FALSE	FALSE	posttest

Table 12 provides results about unwanted web crawlers results in Group 1. The counts, test type, crawler type, group type, and download indicator are depicted in Table 12. In addition to Table 12, a deeper level of detail is provided in Table 13 in terms of what keys and records actually passed or failed. Table 13 contains the results in the same format as Table 12 in terms of number of rows for ease of comparison between two tables. The results indicate that 90 web pages in the pretest step and 90 web pages in the posttest step did not have the five-factor identification keys matched because these keys were not even introduced to this step, as indicated earlier.

Unwanted Web Crawlers Results, Group 2 (Pretest-posttest Treatment Group)

This group has some differences from and similarities to the previous group when it comes to the results and characteristics. The following can be stated about the distinct characteristics of this group. First, this group had exposure to treatment (although the exposure was only limited to posttest process). Second, this group was crawled by unwanted web crawlers similar to the previous group. The unwanted web crawlers' pages for Group 2 consisted of two steps with two results, the pretest and posttest steps, along with results for each step. However, the results for this group were very similar to those of Group 1 but not identical in terms of number of pages downloaded. As depicted in Table 14, unwanted

crawlers attempted to download and crawl to 90 web pages for the pretest and 90 web pages for the posttest step. In terms of the number of successful downloads, only 89 web pages were downloaded in the pretest group, but one web page did not download because of the page not being available on the network. On the other hand, in the posttest group, 90 web pages were crawled and 90 web pages did not download. The results were expected because the five-factor identification was introduced to posttest step. The results for this treatment group are depicted in Table 14:

Table 14

Unwanted Web Crawlers Results, Group 2

Count	Test Type	Crawler Type	Group Type	Downloaded
1	pretest	unwanted	group_2	FALSE
89	pretest	unwanted	group_2	TRUE
90	posttest	unwanted	group_2	FALSE

Table 15

Validation of Five Factorial Keys for Unwanted Web Crawlers, Group 2

COUNT	IP CHECK	PASSKEY CHECK	VISITED CHECK	DATE CHECK	AGENT CHECK	TEST TYPE
89	FALSE	FALSE	FALSE	FALSE	FALSE	pretest
1	FALSE	FALSE	FALSE	FALSE	FALSE	pretest
90	FALSE	FALSE	TRUE	FALSE	TRUE	posttest

Table 15 contains more detailed information about the results pertaining to Group 2 for unwanted web crawlers. The result of the pretest step for this group (including row two with one count) indicates 90 web pages with false values for the IP check, passkey check, visited check, date check, and agent check. The false values are acceptable, and they suggest

that the values did not exist on the web page server and web crawler side in the pretest step. On the other hand, in the posttest step, the results were different as expected because the five-factor identification was introduced during this step. The results for the posttest step of this group indicate that IP check and passkey check returned a false value similar to the pretest step. So three types of keys did not match, but visited check and agent check did return a true value, suggesting that the server keys and web crawler's keys matched. Therefore, the main differences between pretest and posttest results are the values for visited check and agent check.

Wanted Web Crawlers Results Group 1 (Pretest-posttest Control Group)

The two previous groups were designed to capture samples for unwanted web crawlers, but this group contained only the web pages targeted for wanted web crawlers. This group was not exposed to the five-factorial identification treatment because this was a controlled group. In this group, the results were very similar to Group 1 except that the type of web crawler used for this step was different. The pretest result depicted in Table 16 indicates that only one web page did not download, and 89 web pages were downloaded by unwanted web crawlers. Also, the posttest result showed similar results to pretest results because only 89 web pages were downloaded, and one did not download due to unavailable web page error.

Table 16

Wanted Web Crawlers Results Group 1

Count	Test Type	Crawler Type	Group Type	Downloaded
1	pretest	wanted	group_1	FALSE
89	pretest	wanted	group_1	TRUE
1	posttest	wanted	group_1	FALSE
89	posttest	wanted	group_1	TRUE

Table 17

Validation of Five Factorial Keys for Wanted Web Crawlers, Group 1

COUNT	IP CHECK	PASSKEY CHECK	VISITED CHECK	DATE CHECK	AGENT CHECK	TEST TYPE
89	FALSE	FALSE	FALSE	FALSE	FALSE	pretest
1	FALSE	FALSE	FALSE	FALSE	FALSE	pretest
89	FALSE	FALSE	FALSE	FALSE	FALSE	posttest
1	FALSE	FALSE	FALSE	FALSE	FALSE	posttest

The detailed or key level results for this group are depicted in Table 17. The results for this group indicate that the pretest and posttest results were identical in terms of number of counts and IP check, passkey check, visited check, date check, and agent check values, and the keys were all false, indicating that they did not match. Also, there was no difference between the pretest and posttest steps in terms of the results of the keys as depicted in Table 17. The results suggest that all the web pages in rows one and three were downloaded by the web crawler. However, the values for rows two and four in Table 17 did not match the crawler's values because the web pages did not download. The results did not exclude the unavailable web pages because this kind of behavior can also occur on the Internet and World Wide Web. The results in Table 17 match the expected behavior because this group had no exposure to five-factorial identification.

Wanted Web Crawlers Results Group 2 (Pretest-posttest Treatment Group)

The sample Group 1 for wanted web crawler was not exposed to any five-factor identification process, so the main goal for using this group was to have sample web pages for wanted web crawlers. The process of exposing the group to treatment was consistent with previous groups in a way that the posttest was only exposed to treatment. The wanted web

crawlers' web pages had pretest and posttest results similar to those of previous groups, but the process was different in terms of exposure for five-factor identification. The pretest process showed that the web crawler was able to download most of the web pages since 89 web pages out of 90 web pages were downloaded, but one did not download because of unavailable web page error. For the posttest results, the outcome was identical in terms of the number of web pages downloaded or not downloaded by web crawler. Table 18 has more information about count, test type, crawler type, group type, and downloaded results.

Table 18

Wanted Web Crawlers Results Group 2

Count	Test Type	Crawler Type	Group Type	Downloaded
1	pretest	wanted	group_2	FALSE
89	pretest	wanted	group_2	TRUE
1	posttest	wanted	group_2	FALSE
89	posttest	wanted	group_2	TRUE

Table 19

Validation of Five Factorial Keys for Wanted Web Crawlers, Group 2

COUNT	IP CHECK	PASSKEY CHECK	VISITED CHECK	DATE CHECK	AGENT CHECK	TEST TYPE
1	FALSE	FALSE	FALSE	FALSE	FALSE	pretest
89	FALSE	FALSE	FALSE	FALSE	FALSE	pretest
1	FALSE	FALSE	FALSE	FALSE	FALSE	posttest
89	TRUE	TRUE	TRUE	TRUE	TRUE	posttest

The outcome of Group 2 for validation of five-factor identification keys is depicted in Table 19. This group only contained web pages for wanted web crawlers, but unlike Group 1, as indicated earlier, this group was exposed to five-factor identification only during the posttest step. The pretested results showed that the values for IP check, passkey check,

visited check, and agent check were all false, indicating that the keys did not match the server side keys. The pretest group did not have five-factor identification exposure, and it was expected that the values would be false. On the other hand, the posttest results indicated that the values did match for Row Four in Table 19, and only one row did not match as indicated in Row Three for the posttest step.

Classifications for Web Crawlers' Results

One of the goals of this study was to determine whether the five-factor identification process would prevent or allow downloading web pages given unwanted web crawler and wanted web crawler types. However, before actually evaluating the hypothesis, it is critical to make sure that the processed data by SPSS is a correct reflection of actual observed data. The classification tables in this section were created as the result of binary logistic regression output from SPSS. The classification tables in SPSS depict the percentage of correctly predicted value of data based on observed value which SPSS processed, as indicated earlier. This information is another indicator to make sure the processed data by SPSS correctly corresponds to observed data. The easiest way to read the classification tables in this section is from right to left because the most useful information is in the right-most columns. Also, each classification table explains some of the variance for the dependent variable as depicted in Tables 20 and 21. The classification tables in this section provide information about observed and downloaded results for web crawling success or failure processed along with percentage information for success and failure, too. In this section, two classification tables were created because there were two types of web crawlers.

Table 20

Unwanted Web Crawlers Classification Results

Observed			Predicted		
			downloaded		Percentage Correct
			success	failure	
Step 1	downloaded	success	89	0	100.0
		failure	1	90	98.9
	Overall Percentage				99.4

The classification results in Table 20 are about the unwanted web crawlers' classification results, which highlight the number of success and failure observed. This result indicates that from 180 web pages, 89 web pages were downloaded by web crawlers and 91 were not downloaded by the web crawlers. This includes the comparison of control and treatment groups for the unwanted web crawlers only. The "Percentage Correct" column on the right side of Table 18 is simply used to show how successfully SPSS was able to predict the observed values against the observed values. The most important value for Table 20 is the "Overall Percentage" information in the last row, which indicates 99.4% of successful predicted values versus observed values.

Table 21

Wanted Web Crawlers Classification Results

Observed			Predicted		
			downloaded		Percentage Correct
			succeeded	failure	
Step 0	downloaded	success	178	0	100.0
		failure	2	0	.0
	Overall Percentage				98.9

Table 21 provides information about the number of successes and failures of downloads processed by SPSS; however, the main difference between Tables 20 and 21 is the type of web crawler used for the collected data in addition to the outcome differences, which are reflected in each table. *Success* indicates that web crawler was able to download the web page, and *failure* means web crawler was prevented from downloading a web page. In Table 21, the wanted web crawlers attempted to download 190 web pages; 178 web pages were downloaded successfully and two were not. The important number for this table is the "Overall Percentage" data, similar to Table 20. Also, the actual values are different in two tables because the value in Table 21 had 98.9% of correct predicted downloads versus what the observed values were.

Data Reliability

Data validity was an important part of the data analysis step, and in this section the results of data validity are presented. The data validity was done for two groups of data separately. The first group included data related to the wanted web crawlers' download and the result of keys' success or failure. Also, a second group was used to measure unwanted web crawlers as well as the wanted web crawler. So, to increase data validity, unwanted web crawlers' data download and the result of keys' success or failure of download were also captured and measured in this study. In addition, this study used Cronbach's alpha, which is a typical test for various validity analysis for internal reliability evaluation. It basically "calculates the average of all possible split-half reliability coefficients. A computed alpha coefficient will vary between 1 and 0" (Bryman & Bell, 2003). The value 1 indicates a perfect internal reliability, and value 0 indicates no internal reliability (Bryman & Bell, 2003). "The figure 0.80 is typically employed as a rule of thumb to denote an acceptable

level of internal reliability, though many writers accept a slightly lower figure" (Bryman & Bell, 2003). The results of Cronbach's alpha and reliability statistics are provided in Tables 12 and 13.

Table 12

Wanted Web Crawlers

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
.989	.989	2

Cronbach's alpha for wanted web crawlers was .98, which is an acceptable number because 1 indicates a very reliable data and 0 indicates a very unreliable data. The concept of Cronbach's alpha value is widely documented, and what values are acceptable and what values are not are well documented based on the scale of 0-1 (Bryman & Bell, 2003). Table 13 provides reliability statistics information for unwanted web crawlers for this study.

Table 13

Unwanted Web Crawlers

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
.994	.994	2

Cronbach's alpha value for unwanted web crawlers is .99, as depicted in Table 13. This value is very similar to the wanted web crawlers' value. The result for Cronbach's alpha value is valid and acceptable since it is very close to value 1.

Research Questions/Hypotheses Results

In this section, the hypotheses are examined and evaluated to determine whether each hypothesis should be rejected or not rejected. There were two groups of hypotheses for this study, with two hypotheses in each group. The binary logistic regression in SPSS was used to calculate the P-value to see if introducing five-factor identification had any significant effect on the results for control group and treatment group for unwanted web crawler and wanted web crawlers. The results of SPSS analysis for P-value calculations are depicted in Table 22.

Table 22

P-values for Treatment/Intervention Group and Control Group

Type	P-value	Conclusion
unwanted web crawler web pages	0.000	Reject
wanted web crawler web pages	0.097	Do not Reject

The following sections will provide further details about each hypothesis evaluation and how results are used to reject or not reject each hypothesis based on a significant level as it pertains to each type of web crawler. In the following sections, the Group A section contains information about wanted web crawlers, and the Group B section provides information about unwanted web crawlers.

Hypotheses Group A:

Group A hypothesis is designed only for wanted web crawlers and web pages. The two hypotheses in this group provide a framework for evaluation of five-factor identification for web pages in Group 1 (control) and Group 2 (treatment) for wanted web crawlers accessing web pages. The hypotheses in Group A are provided below, and the results are in Table 23.

- ***H_0:*** There is no significant difference between treatment/intervention group and control group in terms of wanted/valid web crawlers visits.
- ***H_1:*** There is a significant difference between treatment/intervention group and control group in terms of wanted/valid web crawlers visits.

Table 23

Outcome Hypotheses Group A

Hypotheses Group A	Outcome
H_0	Do not Reject

The results in Table 23 are based on Binary Logistic Regression and the Omnibus Test. The Omnibus Test is one of the precise statistical methods to determine if "there is a difference between groups (two or more)" (Swanson & Holton, 2005). The outcome of Binary Logistic Regression and the Omnibus Test indicated the P-value of 0.097. The calculated P-Value for wanted web crawlers exceeded the .05 alpha level given the 95% confidence interval. So the outcome of a hypothesis test suggests not rejecting H_0, as depicted in Table 23.

Hypotheses Group B:

The Hypotheses in Group A focused on wanted web crawlers, but the Group B hypotheses are different because they are designed for unwanted web crawlers. The two hypotheses in this group went through a similar process in terms of evaluation of hypotheses. There were two groups: Group 1 as the control and Group 2 as the treatment group.

- ***H_0:*** There is no significant difference between treatment/intervention group and control group in terms of unwanted web crawlers' visits.
- ***H_1:*** There is a significant difference between treatment/intervention group and control group in terms of unwanted web crawlers' visits.

Table 24

Outcome Hypotheses Group B

Hypotheses Group B	Outcome
H_0	Rejected
H_1	Do not Reject

The P-Values in Table 22 were calculated using SPSS Binary Logistic Regression and the Omnibus Test; also, additional information is provided in Appendices A and B. The results of comparing the unwanted web crawler control group and treatment group suggest that there was a significant change since the P-value was less than .05 alpha level, given the 95% confidence interval. So the outcome of a hypothesis test is rejecting H_0 in favor of H_1, as depicted on Table 24.

Summary

Detailed information about the research results was presented in this chapter. The information pertaining to crawlers' return rate results was provided. In addition, demographic characteristics of the sample were described, and tables and graphs were presented. Also, information about how each hypothesis in multiple groups was evaluated based on the statistical analysis results provided, along with how each was rejected or not rejected, was examined and explained. The results of binary logistic regression were provided in addition to explanation and interpretation of the results as they pertained to wanted and unwanted web crawlers.

Chapter 5. Conclusion(s) and Discussion

Introduction

This chapter discusses conclusions based on the statistical testing results and findings about the hypothesis test's outcome. In addition, findings and overall study conclusions pertaining to the five-factor identification process as a defense mechanism against web crawler's intrusion will be explained in detail. Furthermore, the conclusion/discussion section provides information about the implication of five-factor identification of web crawlers in terms of various areas on which future studies need to concentrate, based on the findings of this research.

Conclusion(s) /Discussion

This research examined a novel method to prevent unwanted web crawlers while still allowing valid web crawlers to access web pages. Quantitative measurements and binary logistic regression were used to examine the five-factor identification of web crawlers as a defense mechanism against web crawler intrusion. The results discussed in Chapter 4 provided valuable information to the existing knowledge and resources that have been available for the community of engineers, developers, I.T. specialists, and users by proposing and investigating the use of this new five-factor identification process to prevent unwanted web crawlers' intrusion.

A detailed data collection was completed by using multiple computers and web servers along with web crawlers and web pages. This study examined 720 web pages hosted on 10 servers, with each computer hosting its own dedicated web server for web pages. The web pages were categorized based on visiting web crawler type. Each visiting web crawler was categorized as valid or invalid prior to crawling process. Pretest steps were completed

and results were recorded for data analysis steps. In addition, the posttest was completed after introducing the five-factor identification. The results were collected and stored in data files for traceability and validity; in addition, a database was used to store the results. The results of web crawling were recorded based on success or failure of web crawlers to download a web page. In addition, pass key, date, user agent, IP, and number of visits for the web server/page (allowed each day) was recorded as the five-factor identification keys. The two types of web crawlers were broken in two separate groups for pretest and posttest steps. The groups then were examined for two types of web crawlers to see if using five-factor identification would contribute to preventing unwanted or wanted web crawlers from being able to download web pages. The unwanted web crawlers were labeled Group A, and unwanted web crawlers were labeled Group B.

The statistics and outcomes of binary logistic regression show that by introducing five-factor identification mechanism which included pass key, date, user agent, IP, and number of visits for the web server/page (allowed each day), there was a significant difference between the treatment/intervention group and control group, in terms of unwanted web crawlers visits. This suggests that using five-factor identification contributes to preventing unwanted web crawlers visiting and accessing web pages. The results and findings of this novel solution are critical because various researchers have raised the need to investigate how to identify web crawlers able to prevent the unwanted web crawlers (Stassopoulou & Dikaiakos, 2009; Doran & Gokhale, 2011). Also, many well-known studies have pointed out how web crawlers are misused by unethical entities such as spammers and how this misuse of web crawlers has created ethical, legal, and technical programming challenges (Stassopoulou & Dikaiakos, 2009; Doran & Gokhale, 2011).

Therefore, this study attempted to address the problem and find a solution to some of the earlier documented technical challenges to identify web crawlers. In addition to examining unwanted web crawlers and determining how using five-factor identification may prevent unwanted web crawlers, another group of web pages were constructed, and a dedicated web crawler was used to see how using five-factor identification may inadvertently prevent valid web crawlers. The results and outcome of binary logistic regression indicated that there was no significant difference between the treatment/intervention group and the control group in terms of wanted/valid web crawlers visits. This suggests that deploying and using five-factor identification does not prevent valid web crawlers from accessing or downloading web pages. This finding is important because simply preventing all web crawlers from visiting web pages is not useful and will not reduce web page visibility on search engines such as Google, Bing, and Yahoo, but being able to perform selective exclusion of robots or web crawlers can help better manage web crawlers' visits and work as a gatekeeper to prevent unwanted web crawlers' intrusions for accessing and downloading information from a website without obtaining permission from the owner.

In terms of success rate, the outcome of data analysis suggested that there was 99.4% overall success rate for preventing unwanted web crawlers, and there was a 98.9% success rate for valid web crawlers being able to download web pages even after introducing the five-factor identification (as depicted in Appendix A and B). The overall percentages are also valuable information in terms of confirmation of the findings of this study about the use of five-factor identification to prevent unwanted web crawlers but still allow valid web crawlers to download web pages. The results of this study are compared to proposed solutions of some of the earlier studies, then they suggest the five-factor identification is a very good solution to

prevent unwanted web crawlers from accessing web pages without impacting valid web crawlers because the solution provided in this study addresses some of the weaknesses of earlier proposed solutions. One of the well-known solutions to prevent web crawlers and robots is CAPTCHA, but other researchers have pointed out that CAPTCHA will not be able to protect web crawlers in the near future (Von, Blum, & Langford, 2004). The idea behind CAPTCHA was relatively simple because "colorful images with distorted text in them at the bottom of Web" pages or sites are displayed along with text box (Von, Blum, & Langford, 2004, p. 56). A user would attempt to type those distorted characters into a textbox prior to entering a website. This task can be simple for most people who are not visually impaired, but it can be difficult for those who may have vision problems or hearing problems because some sites provide this mechanism in an audio version. Presenting a distorted image to humans can easily impact user experience and interaction with websites because it is a tedious task for a user to enter some characters into a textbox based on some distorted image; this can discourage some users from even wanting to go to a website. So one of the main drawbacks of CAPTCHA is users' experience, and someone with vision disability will experience challenges. However, five-factor identification does not impact users' experience, and the process is invisible to them. This is a big improvement compared to CAPTCHA because users will not have to change anything when accessing a web page, but five-factor identification will still keep the unwanted web crawlers away.

Another mechanism proposed by earlier researchers is called Clickstream, which is about tracking of user clicks per link, images and buttons (Wang & Lee, 2011). If a web crawler is on a web page, then it attempts to download and crawl to all links on a web page. Therefore, a program on webserver can identify whether a web crawler has entered a web

page. This mechanism works well; however, the main disadvantage is that the mechanism identifies a web crawler after it actually has allowed access to a web crawler to download its content, which is too late. Of course, if a web crawler is identified the first time, then it can be prevented from entering a web page during its next attempt, but even that becomes difficult because web crawlers may not keep the same IP address during multiple crawling sessions. Unlike the Clickstream process, five-factor identification is a more proactive mechanism because it prevents unwanted web crawlers from even accessing the content of web pages.crawler before any downloading occurs. Also, five-factor identification does not rely on any Clickstream patterns to identify a human vs. a web crawler; instead, passkeys are defined between valid web crawlers and a web site hosting web pages, and if any of the five-factor identification keys do not match, then a web crawler will not be allowed to enter a web page.

Another proposed solution by previous researchers, Robots Exclusion Protocol, which uses Robot.txt, is an optional protocol because it does not enforce intended requirements and it cannot keep the integrity of the web host or server when it comes to visibility and access permission of web page. The permission or content access of the web page is defined in a text file based on Robots Exclusion Protocol, and it is valid only if a web crawler decides to follow those guidelines. When it comes to Robots Exclusion Protocol, there are simply no mechanisms to enforce permissions. This problem of lack of enforcement has been well documented by previous researchers, and various studies show that this protocol is not enforced and is ineffective (Sun, Zhuang, & Giles, 2007; Kolay, D'Alberto, Dasdan, & Bhattacharjee, 2008). However, five-factor identification is based on enforcing key validation and forcing web crawlers to provide identification prior to entering a web page. Requiring

web crawlers to provide identity is a great improvement over Robots Exclusion Protocol, which lacks prevention mechanism enforcement when it comes to granting access to a web crawler to visit a web page.

Last, the five-factor identification is not meant to replace all previous proposed solutions such as CAPTCHA but is rather a solution to address some of the weaknesses and drawbacks of previous solutions. So, in short, the five-factor identification can be used along with CAPTCHA or other existing protocols to manage and prevent unwanted web crawlers from accessing, downloading, and consuming web servers' resources. So the outcome of this study should help fill some of the existing gaps in previous solutions such as being able to prevent unwanted web crawlers selectively without impacting valid or acceptable crawlers such as search engine robots and crawlers to access web pages.

Recommendations

The findings and results of this study provided a new mechanism to better prevent and manage unwanted web crawlers, but there are still various paths which were outside of the scope of this study, and it still needs deeper exploration and examination. The following are the recommendations for future studies.

First, this study used only a local area network with 720 web pages to test and implement the five-factor identification mechanisms. A more comprehensive exploration is recommended to explore and implement the five-factor identification mechanism over the World Wide Web on multiple web sites with greater numbers of web pages and web servers.

Second, this study only examined the effect of treatment by comparing the pretest and posttest results for unwanted and wanted web crawlers, but there was no deep or comprehensive examination of the five-factor identification keys in terms of how keys

interact with each other to increase or decrease performance. Exploring how to provide a deeper understanding of most influential keys and how the keys can increase or decrease the web crawler's access and download is suggested for futures studies.

Third, the proposed five-factor identification process only examined the effect of using this mechanism versus web pages and servers that have no web crawlers' management mechanism. A more comprehensive study for future works could be considered in terms of comparing the five-factor identification against some other existing solutions such as Clickstream identification and prevention of web crawlers. This can contribute to the field of software security and web crawler management because it can provide valuable information about the level of effectiveness between uses of five-factor identification and previous proposed solutions.

Fourth, this study focused on the use of five-factor identification and implementing keys on the server side and web crawlers that are interested in obtaining permission and access. However, the handshake or setting up of the keys were manual processes, meaning the keys had to be created ahead of the web crawling process or the web crawler would not have any access to a given web page or site. A future study is needed to provide a solution to automat this handshake and validation ahead of time instead of a manual key setup process.

This study provided a novel mechanism as a way to prevent unwanted web crawlers. However, the field of web crawlers and web security still needs further research, and the suggested recommendations in this section can improve and enable better solutions to prevent unwanted web crawlers without preventing the valid web crawlers such as search engine bots to still access web pages for indexing purposes or any other critical tasks

Summary

In this chapter, conclusions and discussions were provided based on the findings and results of this research. The conclusions and discussions section provided information about how findings of this study are compared to the literature of prior research pertaining to web crawlers. Also, drawbacks and strengths of five-factor identification were examined and compared to various existing mechanisms to manage and prevent unwanted web crawlers such as CAPTCHA and Clickstream from accessing web pages. Furthermore, recommendations were provided to help future works and studies navigate, improve, and concentrate on specific areas of web security and web crawlers' identification and management mechanisms.

References

Artail, H., & Fawaz, K. (2008). A fast HTML web page change detection approach based on hashing and reducing the number of similarity computations. *Data & Knowledge Engineering*, *66*(2), 326 - 337. doi:10.1016/j.datak.2008.04.003

Bai, X., Cambazoglu, B. B., & Junqueira, F. P. (2011). Discovering URLs through user feedback. In *Proceedings of the 20th ACM international conference on Information and knowledge management* (pp. 77-86). New York, NY, USA: ACM. doi:10.1145/2063576.2063592

Banzal, S. (2007). *Data and Computer Network Communication*. Laxmi Publications Pvt Ltd. Retrieved from http://books.google.com/books?id=5xNzTwTGwewC

Barbosa, L., & Freire, J. (2007). An adaptive crawler for locating hidden-Web entry points. In *Proceedings of the 16th international conference on World Wide Web* (pp. 441-450). New York, NY, USA: ACM. doi:10.1145/1242572.1242632

Basarab, D. (2010). *Predictive Evaluation: Ensuring Training Delivers Business and Organizational Results* (First.). Berrett-Koehler Publishers. Retrieved from http://common.books24x7.com.ezproxy.emich.edu/toc.aspx?bookid=41248

Bewick, V., Cheek, L., & Ball, J. (2005). Statistics review 14: Logistic regression. *Critical Care*, *9*(1), 112-118. doi:10.1186/cc3045

Blank, G., Fielding, N. G., & Lee, R. M. (2008). *The SAGE Handbook of Online Research Methods*. SAGE Publications. Retrieved from http://books.google.com/books?id=EeMKURpicCgC

Branzburg, J. (2007, February). Lean and mean: keep your PC running smoothly with these tips. *Technology & Learning*, *27*(7), 28+.

Brown, R. (2009). *Public Relations and the Social Web: How to Use Social Media and Web 2.0 in Communications*. Kogan Page. Retrieved from http://books.google.com/books?id=b6zZAAAAMAAJ

Bryman, A., & Bell, E. (2003). *Business research methods*. Oxford University Press, Incorporated. Retrieved from http://library.books24x7.com.ezproxy.emich.edu/assetviewer.aspx?bookid=12878&chunkid=182510142

Cafarella, M. J., Halevy, A., & Madhavan, J. (2011). Structured data on the web. *Commun. ACM*, *54*(2), 72-79. doi:10.1145/1897816.1897839

Cai, R., Yang, J.-M., Lai, W., Wang, Y., & Zhang, L. (2008). iRobot: an intelligent crawler for web forums. In *Proceedings of the 17th international conference on World Wide Web* (pp. 447-456). New York, NY, USA: ACM. doi:10.1145/1367497.1367558

Campbell, D., & Stanley, J. (1973). *Experimental and Quasi-experimental Designs for Research*. Rand McNally. Retrieved from http://books.google.com/books?id=2zQfAQAAIAAJ

Chandramouli, A., & Gauch, S. (2007). A co-operative web services paradigm for supporting crawlers. In *Large Scale Semantic Access to Content (Text, Image, Video, and Sound)* (pp. 475-489). Paris, France, France: LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE. Retrieved from http://dl.acm.org/citation.cfm?id=1931390.1931437

Chen, L., Bhowmick, S. S., & Nejdl, W. (2009). NEAR-Miner: mining evolution associations of web site directories for efficient maintenance of web archives. *Proc. VLDB Endow.*, *2*(1), 1150-1161.

Chiang, R. H. L., Goes, P., & Stohr, E. A. (2012). Business Intelligence and Analytics Education, and Program Development: A Unique Opportunity for the Information Systems Discipline. *ACM Trans. Manage. Inf. Syst.*, *3*(3), 12:1-12:13. doi:10.1145/2361256.2361257

Cho, J., & Garcia-Molina, H. (2003). Effective page refresh policies for Web crawlers. *ACM Trans. Database Syst.*, *28*(4), 390-426. doi:10.1145/958942.958945

Coleman, P., & Nelson, S. (2000). *Effective Executive's Guide to the Internet: The Seven Core Skills Required to Turn the Internet Into a Business Power Tool*. Course Technology Ptr. Retrieved from http://books.google.com/books?id=y0jBSgAACAAJ

Compart, A. (2009). Ryanair Gives Web Site Fare Access But Continues Fight With Others. *Aviation Daily*, *376*(42), 5.

Dalvi, N., Machanavajjhala, A., & Pang, B. (2012). An analysis of structured data on the web. *Proc. VLDB Endow.*, *5*(7), 680-691.

Dikaiakos, M., Stassopoulou, A., & Papageorgiou, L. (2003). Characterizing Crawler Behavior from Web Server Access Logs. In K. Bauknecht, Am. Tjoa, & G. Quirchmayr (Eds.), *E-Commerce and Web Technologies* (Vol. 2738, pp. 369-378). Springer Berlin Heidelberg. Retrieved from http://dx.doi.org/10.1007/978-3-540-45229-4_36

Dittrich, D., Bailey, M., & Dietrich, S. (2009). *Towards Community Standards for Ethical Behavior in Computer Security Research*. Retrieved from http://staff.washington.edu/dittrich/papers/dbd2009tr1-20090925-1133.pdf

Divanna, J. (2003). *Thinking Beyond Technology: Creating New Value in Business*. New York, NY: Palgrave Macmillan. Retrieved from http://books.google.com/books?id=Ti0_mQEACAAJ

Doran, D., & Gokhale, S. S. (2011). Web robot detection techniques: overview and limitations. *Data Mining and Knowledge Discovery*, *22*(1-2), 183-210.

Douglis, F., Feldmann, A., Krishnamurthy, B., & Mogul, J. (1997). Rate of change and other metrics: a live study of the world wide web. In *Proceedings of the USENIX Symposium on Internet Technologies and Systems on USENIX Symposium on Internet Technologies and Systems* (pp. 14-14). Berkeley, CA, USA: USENIX Association. Retrieved from http://dl.acm.org/citation.cfm?id=1267279.1267293

Edwards, J., McCurley, K., & Tomlin, J. (2001). An adaptive model for optimizing performance of an incremental web crawler. In *Proceedings of the 10th international conference on World Wide Web* (pp. 106-113). New York, NY, USA: ACM. doi:10.1145/371920.371960

Ehrig, M., & Maedche, A. (2003). Ontology-focused crawling of Web documents. In *Proceedings of the 2003 ACM symposium on Applied computing* (pp. 1174-1178). New York, NY, USA: ACM. doi:10.1145/952532.952761

Eichmann, D. (1995). Ethical Web agents. *Computer Networks and ISDN Systems*, *28*, 127 - 136. doi:10.1016/0169-7552(95)00107-3

Feldman, M. P. (2002). The Internet revolution and the geography of innovation. *International Social Science Journal*, *54*(171), 47-56.

Fraternali, P. (1999). Tools and approaches for developing data-intensive Web applications: a survey. *ACM Comput. Surv.*, *31*(3), 227-263. doi:10.1145/331499.331502

Gangadharan, V. P., & Pretorius, L. (2010). Towards an ethical analysis of the W3C Web services architecture model. In *Information Security for South Africa*. doi:10.1109/ISSA.2010.5588642

Giles, C. L., Sun, Y., & Councill, I. G. (2010). Measuring the web crawler ethics. In *Proceedings of the 19th international conference on World wide web* (pp. 1101-1102). New York, NY, USA: ACM. doi:10.1145/1772690.1772824

Google. (2012a, February 17). Robots meta tag and X-Robots-Tag HTTP header specifications - Webmasters. Google. Retrieved from https://developers.google.com/webmasters/control-crawl-index/docs/robots_meta_tag

Google. (2012b, June 13). Customizing Results Snippets. Google. Retrieved from https://developers.google.com/custom-search/docs/snippets#creating_snippets

Google. (2012c, September 24). Googlebot Webmaster Tools. Google. Retrieved from http://support.google.com/webmasters/bin/answer.py?hl=en&answer=182072

Govardhan, A., Narayana, V. A., & Premchand, P. (2009). Effective detection of near duplicate web documents in web crawling. *International Journal of Computational Intelligence Research*, *5*, 83+.

Isaac, S., & Michael, W. (1981). *Handbook in research and evaluation: a collection of principles, methods, and strategies useful in the planning, design, and evaluation of studies in education and the behavioral sciences*. EDITS Publishers. Retrieved from http://books.google.com/books?id=yEB-AAAAMAAJ

Ke, Y., Deng, L., Ng, W., & Lee, D.-L. (2006). Web dynamics and their ramifications for the development of Web search engines. *Computer Networks*, *50*(10), 1430 - 1447. doi:10.1016/j.comnet.2005.10.012

Kleinbaum, D. G., & Klein, M. (2010). *Logistic Regression: A Self-Learning Text*. New York, NY: Springer Science+Business Media, LLC,. Retrieved from http://portal.emich.edu/vwebv/holdingsInfo?bibId=1055152

Koehl, A., & Wang, H. (2012). Surviving a search engine overload. In *Proceedings of the 21st international conference on World Wide Web* (pp. 171-180). New York, NY, USA: ACM. doi:10.1145/2187836.2187860

Kogut, B. M. (2004). *The Global Internet Economy*. Mit Press. Retrieved from http://books.google.com/books?id=KS3IPQbeINcC

Kolay, S., Dalberto, P., Dasdan, A., & Bhattacharjee, A. (2008). A larger scale study of robots.txt. In *Proceedings of the 17th international conference on World Wide Web* (pp. 1171-1172). New York, NY, USA: ACM. doi:10.1145/1367497.1367711

Koster, M. (1995). Robots in the Web: Threat or Treat? *ConneXions*, *9*(4), 1-8.

Krishnamurthy, B., Mogul, J. C., & Kristol, D. M. (1999). Key differences between HTTP/1.0 and HTTP/1.1. *Computer Networks*, *31*(1-16), 1737 - 1751. doi:10.1016/S1389-1286(99)00008-0

Kumar, M., & Vig, R. (2009). Design of CORE: context ontology rule enhanced focused web crawler. In *Proceedings of the International Conference on Advances in Computing, Communication and Control* (pp. 494-497). New York, NY, USA: ACM. doi:10.1145/1523103.1523201

Kuusisto, F. (2012). XRDS: Crossroads, The ACM Magazine for Students - The Role of Academia in the Startup World. *XRDS*, *18*(4), 41. doi:10.1145/2173637.2173654

Ledford, J. (2007). *SEO: Search Engine Optimization Bible*. John Wiley & Sons. Retrieved from http://books.google.com/books?id=sgmxo1Alq_4C

Lee, H.-T., Leonard, D., Wang, X., & Loguinov, D. (2008). IRLbot: scaling to 6 billion pages and beyond. In *Proceedings of the 17th international conference on World Wide Web* (pp. 427-436). New York, NY, USA: ACM. doi:10.1145/1367497.1367556

Leedy, P. D., & Ormrod, J. E. (2005). *Practical Research: Planning and Design* (8th ed.). Upper Saddle River, NJ: Prentice Hall.

Likarish, P., & Jung, E. (2009). A targeted web crawling for building malicious javascript collection. In *Proceedings of the ACM first international workshop on Data-intensive software management and mining* (pp. 23-26). New York, NY, USA: ACM. doi:10.1145/1651309.1651317

Lin, J.-L. (2009). Detection of cloaked web spam by using tag-based methods. *Expert Systems with Applications*, *36*(4), 7493 - 7499. doi:10.1016/j.eswa.2008.09.056

Lourenco, A. G., & Belo, O. O. (2006). Catching web crawlers in the act. In *Proceedings of the 6th international conference on Web engineering* (pp. 265-272). New York, NY, USA: ACM. doi:10.1145/1145581.1145634

Madhavan, J., Ko, D., Kot, L., Ganapathy, V., Rasmussen, A., & Halevy, A. (2008). Google's Deep Web crawl. *Proc. VLDB Endow.*, *1*(2), 1241-1252. doi:10.1145/1454159.1454163

Mali, S., & Meshram, B. B. (2011). Focused web crawler with revisit policy. In *Proceedings of the International Conference & Workshop on Emerging Trends in Technology* (pp. 474-479). New York, NY, USA: ACM. doi:10.1145/1980022.1980125

Mao, Z., & Herley, C. (2011). A robust link-translating proxy server mirroring the whole web. *SIGAPP Appl. Comput. Rev.*, *11*(2), 30-42. doi:10.1145/1964144.1964149

McClave, J. T., Benson, P. G., & Sincich, T. (2001). *Statistics for Business and Economics: Books a La Carte Edition* (8th ed.). New Jersey: Prentice Hall.

McCown, F., & Nelson, M. L. (2006). Evaluation of crawling policies for a web-repository crawler. In *Proceedings of the seventeenth conference on Hypertext and hypermedia* (pp. 157-168). New York, NY, USA: ACM. doi:10.1145/1149941.1149972

Menczer, F., Pant, G., & Srinivasan, P. (2004a). Topical web crawlers: Evaluating adaptive algorithms. *ACM Trans. Internet Technol.*, *4*(4), 378-419. doi:10.1145/1031114.1031117

Menczer, F., Pant, G., & Srinivasan, P. (2004b). Topical web crawlers: Evaluating adaptive algorithms. *ACM Trans. Internet Technol.*, *4*(4), 378-419. doi:10.1145/1031114.1031117

Menczer, F., Pant, G., Srinivasan, P., & Ruiz, M. E. (2001). Evaluating topic-driven web crawlers. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 241-249). New York, NY, USA: ACM. doi:10.1145/383952.383995

Microsoft. (2012). Meet our crawlers. Microsoft. Retrieved from http://onlinehelp.microsoft.com/en-us/bing/hh204496.aspx

Mowery, D. C., & Simcoe, T. (2002). Is the Internet a US invention?-an economic and technological history of computer networking. *Research Policy*, *31*(8-9), 1369 - 1387. doi:10.1016/S0048-7333(02)00069-0

Nelson, S., & Coleman, P. (2000). *Mba's Guide to the Internet: The Essential Internet Reference for Business Professionals*. Redmond Technology Press. Retrieved from http://books.google.com.au/books?id=zz0LAAAACAAJ

Ntoulas, A., Zerfos, P., & Cho, J. (2005). Downloading textual hidden web content through keyword queries. In *Proceedings of the 5th ACM/IEEE-CS joint conference on Digital libraries* (pp. 100-109). New York, NY, USA: ACM. doi:10.1145/1065385.1065407

Oktay, H., Taylor, B., & Jensen, D. (2010). Causal Discovery in Social Media Using Quasi-Experimental Designs.

Pant, G., & Srinivasan, P. (2005). Learning to crawl: Comparing classification schemes. *ACM Trans. Inf. Syst.*, *23*(4), 430-462. doi:10.1145/1095872.1095875

Pant, G., Tsioutsiouliklis, K., Johnson, J., & Giles, C. L. (2004). Panorama: extending digital libraries with topical crawlers. In *Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries* (pp. 142-150). New York, NY, USA: ACM. doi:10.1145/996350.996384

Park, K., Pai, V. S., Lee, K.-W., & Calo, S. (2006). Securing web service by automatic robot detection. In *Proceedings of the annual conference on USENIX 06 Annual Technical Conference* (pp. 23-23). Berkeley, CA, USA: USENIX Association. Retrieved from http://dl.acm.org/citation.cfm?id=1267359.1267382

Perugini, S. (2008). Symbolic links in the Open Directory Project. *Information Processing & Management*, *44*(2), 910 - 930. doi:10.1016/j.ipm.2007.06.005

Pew, R. W., & Van Hemel, S. B. (2004). *Technology for Adaptive Aging.* National Academies Press. Retrieved from http://ezproxy.emich.edu/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=109204&site=ehost-live&scope=site

Qureshi, A. A. (2006). *Network intrusion detection using an innovative statistical approach*. Retrieved from http://ezproxy.emich.edu/login?url=http://search.proquest.com/docview/304960273?accountid=10650

Radhakishan, V., Farook, Y., & Selvakumar, S. (2010). CRAYSE: design and implementation of efficient text search algorithm in a web crawler. *SIGSOFT Softw. Eng. Notes*, *35*(4), 1-8. doi:10.1145/1811226.1811236

Ramirez, J., & Ramirez, B. (2009). *Analyzing and Interpreting Continuous Data Using Jmp: A Step-By-Step Guide*. Sas Inst. Retrieved from http://books.google.com/books?id=H8YiTo8gNU8C

Rao, R., & Vrudhula, S. (2007). Energy optimal speed control of a producer-consumer device pair. *ACM Trans. Embed. Comput. Syst.*, *6*(4). doi:10.1145/1274858.1274868

Ratner, B. (2012). *Statistical and machine-learning data mining: techniques for better predictive modeling and analysis of big data* (Second.). Auerbach Publications. Retrieved from http://common.books24x7.com.ezproxy.emich.edu/toc.aspx?bookid=46728

Ruffer, S. M., Yen, D., & Lee, S. (1995). Client/server computing technology: A framework for feasibility analysis and implementation. *International Journal of Information Management*, *15*(2), 135 - 150. doi:10.1016/0268-4012(95)00008-U

Salem, A. M. (2001). *A software testing model: Using Design of Experiments (DOE) and logistic regression*. Retrieved from http://ezproxy.emich.edu/login?url=http://search.proquest.com/docview/251688330?accountid=10650

Sathyan, J. (2010). *Fundamentals of EMS, Nms and OSS/BSS*. Taylor & Francis. Retrieved from http://books.google.com/books?id=7vv1PQAACAAJ

Schlotzhauer, S. D. (2009). *Elementary Statistics Using SAS*. Sas Inst. Retrieved from http://books.google.com/books?id=_-2V1o9EDrcC

Schrader, M., Vlamis, D., Nader, M., Claterbos, C., Collins, D., Campbell, M., & Conrad, F. (2010). *Oracle Essbase & Oracle OLAP: The Guide to Oracle's Multidimensional Solution*. McGraw-Hill Companies,Incorporated. Retrieved from http://common.books24x7.com.ezproxy.emich.edu/toc.aspx?bookid=33519

Shahriar, H., & Zulkernine, M. (2012). Mitigating program security vulnerabilities: Approaches and challenges. *ACM Comput. Surv.*, *44*(3), 11:1-11:46. doi:10.1145/2187671.2187673

Sharma, D. K., & Sharma, A. K. (2011). A QIIIEP based domain specific hidden web crawler. In *Proceedings of the International Conference & Workshop on Emerging Trends in Technology* (pp. 224-227). New York, NY, USA: ACM. doi:10.1145/1980022.1980073

Sheather, S. J. (2009). Logistic Regression. In *A Modern Approach to Regression with R* (pp. 263-303). Springer New York. Retrieved from http://dx.doi.org/10.1007/978-0-387-09608-7_8

Sosinsky, B. (2009). *Networking Bible*. John Wiley & Sons. Retrieved from http://books.google.com/books?id=0rpeVz-wECwC

Stassopoulou, A., & Dikaiakos, M. D. (2009). Web robot detection: A probabilistic reasoning approach. *Computer Networks*, *53*(3), 265 - 278. doi:10.1016/j.comnet.2008.09.021

Steed, A., & Oliveira, M. F. (2009). *Networked Graphics: Building Networked Games and Virtual Environments*. Elsevier Science. Retrieved from http://books.google.com/books?id=76C_quJqVXcC

Stephens, L. (2004). *Advanced Statistics Demystified*. McGraw-Hill Companies,Incorporated. Retrieved from http://books.google.com/books?id=Aw6ZA-LJ6jQC

Sun, Y. (2008). *A comprehensive study of the regulation and behavior of web crawlers*. Retrieved from http://ezproxy.emich.edu/login?url=http://search.proquest.com/docview/231557647?accountid=10650

Sun, Y., Councill, I. G., & Giles, C. L. (2010). The Ethicality of Web Crawlers. In *Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01* (pp. 668-675). Washington, DC, USA: IEEE Computer Society. doi:10.1109/WI-IAT.2010.316

Sun, Y., Zhuang, Z., Councill, I. G., & Giles, C. L. (2007). Determining bias to search engines from robots.txt. In *IEEE/WIC/ACM INTERNATIONAL CONFERENCE ON WEB INTELLIGENCE* (pp. 149-155). IEEE Computer Society. doi:10.1109/WI.2007.45

Sun, Y., Zhuang, Z., & Giles, C. L. (2007). A large-scale study of robots.txt. In *Proceedings of the 16th international conference on World Wide Web* (pp. 1123-1124). New York, NY, USA: ACM. doi:10.1145/1242572.1242726

Swanson, R., & Holton, E. (2005). *Research In Organizations: Foundations And Methods Of Inquiry*. Berrett-Koehler Publishers, Incorporated. Retrieved from http://library.books24x7.com.ezproxy.emich.edu/assetviewer.aspx?bookid=11859&chunkid=376546207

Taboada, G., Ramos, S., Exposito, R., Tourino, J., & Doallo, R. (2011). Java in the High Performance Computing arena: Research, practice and experience. *Science of Computer Programming*, (0), -. doi:10.1016/j.scico.2011.06.002

Tan, P.-N., & Kumar, V. (2002). Discovery of Web Robot Sessions Based on their Navigational Patterns. *Data Mining and Knowledge Discovery, 6*(1), 9-35.

Thelwall, M., & Stuart, D. (2006). Web crawling ethics revisited: Cost, privacy, and denial of service. *Journal of the American Society for Information Science and Technology, 57*(13), 1771-1779. doi:10.1002/asi.20388

Tsai, C.-F. (2002). A network processing model for address learning and IP recognition. *Information Sciences, 147*, 267 - 280. doi:10.1016/S0020-0255(02)00295-5

Tuffery, S. (2011). *Data Mining and Statistics for Decision Making*. John Wiley & Sons. Retrieved from http://books.google.com/books?id=f030h4MYOJsC

University of California. (2000). How much information. *Internet*. Education. Retrieved January 11, 2013, from http://www2.sims.berkeley.edu/research/projects/how-much-info/internet.html

Von Ahn, L., Blum, M., & Langford, J. (2004). Telling humans and computers apart automatically. *Commun. ACM*, *47*(2), 56-60. doi:10.1145/966389.966390

Wang, D. Y., Savage, S., & Voelker, G. M. (2011). Cloak and dagger: dynamics of web search cloaking. In *Proceedings of the 18th ACM conference on computer and communications security* (pp. 477-490). New York, NY, USA: ACM. doi:10.1145/2046707.2046763

Wang, F. F. (2006). Domain names management and legal protection. *International Journal of Information Management*, *26*(2), 116 - 127. doi:10.1016/j.ijinfomgt.2005.11.003

Wang, X. (2006). Exploring sample sizes for content analysis of online news sites. In *Exploring sample sizes for content analysis of online news sites*. Presented at the Communication Theory & Methodology Division, Association for Education in Journalism and Mass Communication. Retrieved from http://www1.usfsp.edu/journalism/showcase/documents/wangSampleSizesPaper.pdf

Wang, Y.-T., & Lee, A. J. T. (2011). Mining Web navigation patterns with a path traversal graph. *Expert Systems with Applications*, *38*(6), 7112 - 7122. doi:10.1016/j.eswa.2010.12.058

Watson, M. (2009). *Scripting Intelligence: Web 3.0 Information Gathering and Processing*. Apress. Retrieved from http://books.google.com/books?id=ElSxSLsc3M0C

Westfall, R. (2009). If your pearls of wisdom fall in a fores. *Commun. ACM*, *52*(11), 146-149. doi:10.1145/1592761.1592795

Wills, C. E., & Mikhailov, M. (1999). Towards a better understanding of Web resources and server responses for improved caching. In *Proceedings of the eighth international conference*

on World Wide Web (pp. 1231-1243). New York, NY, USA: Elsevier North-Holland, Inc. Retrieved from http://dl.acm.org/citation.cfm?id=313234.313021

Wise, C. (2007). Using Semantic, Structured HTML to Create Web Pages. In *Foundations of Microsoft Expression Web* (pp. 83-106). Apress. Retrieved from http://dx.doi.org/10.1007/978-1-4302-0392-6_4

Wu, B., & Davison, B. D. (2006). Detecting semantic cloaking on the web. In *Proceedings of the 15th international conference on World Wide Web* (pp. 819-828). New York, NY, USA: ACM. doi:10.1145/1135777.1135901

Yahoo. (2011, December 1). Content quality guidelines. Yahoo. Retrieved from http://help.yahoo.com/kb/index?locale=en_US&y=PROD_ACCT&page=content&id=SLN2245

Yalcin, N., & Kose, U. (2010). What is search engine optimization: SEO? *Procedia - Social and Behavioral Sciences*, *9*(0), 487 - 493. doi:10.1016/j.sbspro.2010.12.185

Zabriskie, J. F. (2009). Bots, Scrapers, and Other Unwanted Visitors to Your Web Site: Can You Keep Them Out? *Computer and Internet Lawyer*, *26*(7), 5-11.

Zhong, T. (2010). *An Enhanced Malicious Web Crawler Detection and Classification System*. Retrieved from http://ezproxy.emich.edu/login?url=http://search.proquest.com/docview/787900915?accountid=10650

APPENDIX A: Binary Logistic Regression Results - Unwanted Web Crawlers

Block 1: Method = Enter

Omnibus Tests of Model Coefficients

		Chi-square	df	Sig.
Step 1	Step	238.522	1	.000
	Block	238.522	1	.000
	Model	238.522	1	.000

Model Summary

Step	-2 Log likelihood	Cox & Snell R Square	Nagelkerke R Square
1	10.988[a]	.734	.979

a. Estimation terminated at iteration number 20 because maximum iterations has been reached. Final solution cannot be found.

Classification Table[a]

Observed			Predicted		
			downloaded		Percentage Correct
			success	failure	
Step 1	downloaded	success	89	0	100.0
		failure	1	90	98.9
	Overall Percentage				99.4

APPENDIX B: Binary Logistic Regression Results - wanted Web Crawlers

Block 1: Method = Enter

Omnibus Tests of Model Coefficients

		Chi-square	df	Sig.
Step 1	Step	2.750	1	.097
	Block	2.750	1	.097
	Model	2.750	1	.097

Model Summary

Step	-2 Log likelihood	Cox & Snell R Square	Nagelkerke R Square
1	19.227[a]	.015	.132

a. Estimation terminated at iteration number 20 because maximum iterations has been reached. Final solution cannot be found.

Classification Table[a]

Observed			Predicted		
			downloaded		Percentage Correct
			success	failure	
Step 1	downloaded	success	178	0	100.0
		failure	2	0	.0
	Overall Percentage				98.9

a. The cut value is .500