Web Crawlers Detection

Yomna ElRashidy
yomna.el-rashidi@aucegypt.edu
Introduction
The need for web crawlers detection
Web crawlers methodology
State of the art in web crawlers detection methodologies
  - Using analysis of web logs
  - Using web page member lists
  - Using robot traps strategies
  - Using statistical analysis
Current limitations and future work
Introduction

A web crawler is a program that traverse the web autonomously with the purpose of discovering and retrieving content and knowledge from the Web on behalf of various Web-based systems and services.

For example: search engine crawlers seek to harvest as much Web content as possible on a regular basis, in order to build and maintain large search indexes.
The Need For Web Crawlers Detection

- The amount of traffic caused by crawlers may result to performance degradation of busy web servers.
- Content delivery web sites may not wish to serve incoming HTTP requests from unauthorized web crawlers.
- In human-user profiling using data mining of log files, requests originating from crawlers may provide misleading results regarding the navigational patterns of real users.
- Pay-per-click advertising can be seriously harmed by click fraud which involves among other things the unwilling or malicious repetitive "clicking" on advertising links by Web robots.
Web Crawlers Methodology

A Web crawler is one type of robot, or software agent. It starts with a list of URLs to visit and as the crawler visits these URLs, it identifies all the hyperlinks for the page’s image files, script files, CSS, etc belonging to the requested URL. It adds those hyperlinks to the list of URLs to visit. Those URL’s are recursively visited according to a set of policies and their content is downloaded.
State Of The Art

- Web crawlers detection methodologies
  - Using robot traps strategy
  - Using web page member lists
  - Using analysis of web logs
  - Using statistical analysis
Detection Using Robot Traps Strategy

The foundation of this solution is the use of the robots.txt protocol that indicates which files that are not to be accessed by web crawlers through a set of rules.

Invalid crawlers would fetch files regardless of the rules stated in the robots.txt file.

Distinguishing invalid crawlers by:
- Placing invisible “web-bug” in web pages that link to locations that are restricted within robots.txt
- Web crawlers that visit those locations anyway are identified to be malicious crawlers and their IPs get blocked.

User-agent: *
Disallow: /site=
Disallow: /cite
Disallow: /5480.iac.
Disallow: /go/
Disallow: /audio.html/
Disallow: /houseads/
Disallow: /askhome/
Disallow: /t20pt/
Allow: /
Sitemap: http://www.thesaurus.com/thesaurus-sitemap/Sitemap.xml
Detection Using Web Page’s Member Lists

**Difference Between Crawlers Visit And Human Visit**

**On HTML document request, browser analyses and request all embedded and linked objects to the requested document, such as CSS, image/audio files, script files, etc.**

The browser requests the embedded objects at once within 0-5 seconds, where the total requests intervals never exceeds 30 seconds.

**On HTML document request, crawler analyzes all embedded and linked objects to the requested document such as CSS, image/audio files, script files, etc.**

The crawler doesn’t request linked objects at once and some crawlers add them to waiting lists. The time interval for the requests is greater than 30 seconds.
Member lists are constructed for every page with all its linked objects.

Algorithm analyzes web logs data for every visitor and constructs ShowTable as shown.

ShowNumber values within the table are incremented according to whether the objects within its member list was requested within 30 seconds or not.

A visitors’ ShowTable whose all ShowNumber entries are all zeros, indicate a web crawler rather than a human visitor.
Detection Using Web Logs Analysis

Web Logs Analysis Steps

- “Preprocessing log files” which pertains information about web access attempts such as client IP address, date and time, status code, etc.
- “Session identification”, done by grouping HTTP requests in the log files into sessions. Grouping based on IP address. Where sessions within a certain timeout period are grouped together, otherwise a new session group is created for the same IP address if timeout period is exceeded.
- “Crawler identification” based on certain features:
  - Access of robot.txt
  - Access of hidden links
  - Blank referrer with hit count
  - Hit count
Web crawlers have to **access the robot.txt** before downloading any content from a website.

Introducing hidden links that are not visible in the browser as a honeypot for web crawlers. Sessions that **access the hidden links** are web crawlers.

Crawlers initiate HTTP requests with an **unassigned referrer**, which is used for identification in conjunction with **hit count** exceeding a certain threshold. **Since many browsers exclude referrer which might confuse normal user with a crawler.**

**Hit count** per a particular period, which is the number of HTTP requests during each session. If hit count exceeds a certain threshold, a web crawler is detected.
Detection using statistical analysis

Detecting web crawlers in real-time using machine learning techniques. It utilizes an off-line, probabilistic web crawler detection system, in order to characterize crawlers and extract their most discriminating features based on statistical analysis to be used in detection.

Algorithm overview:

● Characterization of crawler sessions using off-line detection:
  ○ Access-logs analysis and session identification.
  ○ Extraction of session features to be used in the Bayesian network.
  ○ Learning Bayesian network parameters.
  ○ Classification of sessions into crawlers or humans.
● New Features extraction from the classified sessions.
● Statistical Analysis of those features.
● Selection of the most discriminant features to be used by the final real-time detection system.
Detection using statistical analysis

**Initial extracted features from log analysis**

- **Percentage of head requests,** where head requests are done by crawlers to retrieve pages meta data.
- Percentage of 2xx, 3xx responses, higher percentages in human behavior.
- Percentage of page requests.
- Percentage of night requests.
- Average time between requests.
- Standard deviation of time between requests.

**Most discriminant features**

- Percentage of Image requests.
- Percentage of page requests.
- Percentage of 4xx response code.
- Session timeout threshold.
- Maximum click rate.

---

Bayesian network → Statistical Analysis → Realtime Crawler Detection
Limitations Of Current Detection Methodologies

- Methodologies that rely on the fact that web pages usually have other linked resources (page’s member list) such as images, style sheets or scripts, may regard a human as a crawler.
  - If the web pages of a web site contain only plain text.
  - If user uses simple browser or set the browser to not display anything but text.
- Methodologies that rely on session’s behaviour such as percentage of requests or time between requests, requires logging a certain number of requests per session to identify the requesting entity as a human or crawler.
- Methodologies that rely on the request logs for identifying the requesting entity, perform their analysis on log records in offline environments, does not detect crawlers on real time.
- Methodologies that rely on using statistical techniques for real time identification provides average precision of 86%.
Future Work

○ Methodologies that does not rely on past logging records for defining users behaviours should be furtherly researched to allow real time crawlers detection. Further investigations should include adding honeypots for crawlers in the form of hidden links for instance.

○ Real time methodologies that rely on machine learning techniques requires further enhancements to improve levels of precision and accuracy.
References

- **Identification and characterization of crawlers through analysis of web logs**
  Algiriyage, N.; Univ. of Moratuwa, Moratuwa, Sri Lanka; Jayasena, S.; Dias, G.; Perera, A.
  8th IEEE International Conference on Industrial and Information Systems (ICIIS), 2013

- **A Web Crawler Detection Algorithm Based on Web Page Member List**
  Weigang Guo; Sch. of Electron. & Inf. Eng., Foshan Univ. Foshan, Foshan, China; Yong Zhong; Jianqin Xie
  4th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2012

- **PUBCRAWL: protecting users and businesses from CRAWLers**
  roire Jacob, Engin Kirda, Christopher Kruege, Giovanni Vigna

- **Real-time web crawler detection**
  Balla, A. Dept. of Comput. Sci., Univ. of Cyprus, Nicosia, Cyprus
  Stassopoulou, A.; Dikaiakos, M.D.
  18th International Conference on Telecommunications (ICT), 2011

- **A research on a defending policy against the Webcrawler's attack**
  Wei Tong; Sch. of Comput. Sci. & Technol., Guizhou Univ., Guiyang, China; Xiaoyao Xie
Thank you

Yomna ElRashidy
yomna.el-rashidi@aucegypt.edu