The Nuts and Bolts of a Forum Spam Automator

Youngsang Shin, Minaxi Gupta, Steven Myers
School of Informatics and Computing, Indiana University - Bloomington
shiny@cs.indiana.edu, minaxi@cs.indiana.edu, samyers@indiana.edu
Motivation

- The Web is huge and keeps expanding
  - Over **255 million** active websites on the Internet
    - **21.4 million** were newly added in 2010
  - Google claimed to know of one trillion pages even in 2008
- **Making a website discoverable is challenging!**
- **Web spamming**
  - Exploiting Search Engine Optimization (SEO) techniques
    - Keyword stuffing, cloaking
    - Link farms
    - Content farms
Why Forum Spamming?

- Forum
  - A website where visitors can contribute content
  - Examples
    - Web boards, blogs, wikis, guestbooks
- Forums are an attractive target for spamming
  - Many contain valuable information
  - Blacklisting or taking-down is not an option in most cases
- Spammers’ benefit from forum spamming
  - Visitors could be directed to spammers’ websites
  - Boosting search engine rankings for their websites
Overview of Forum Spam Automators

- **Basic function**
  - To automate the process of posting forum spam

- **Advanced Functions**
  - **Goal:** to improve the success rate of spamming
  - **Approach:** to avoid forum spam mitigation techniques
    - Registration
    - Email address verification
    - Legitimate posting history
    - CAPTCHA

- **Examples**
  - **XRumer**, SEnuke, ScrapeBox, AutoPligg, Ultimate WordPress Comment Submitter (UWCS)
Outline

- Introduction
- Overview of Forum Spam Automators
- Primal Functionalities
- Advanced Functionalities
- Traffic Characteristics
- Comparison among Forum Spam Automators
- Conclusion
Primal Functionalities 1/2

- Collecting target forums: *Hrefer*
  - Keywords: *Google AdWords* *Keyword Tool*
  - Search engines: Google, Google Blog Search, MSN, Yahoo, AltaVista, Yandex

- Composing spam messages
  - Various *macros* for composing spam semantically similar but syntactically different spam messages
Primal Functionalities 2/2

Posting Spam

- Supports multiple forum platforms
  - Unknown forum platforms can be learned

Registration

Posting

- Priority categorization to determine topic or discussion to post to
Advanced Functionalities 1/2

- Solving CAPTCHAs
  - Manual mode
  - Automatic mode: solving simple types of CAPTCHAs
    - Question-based & graphic-based CAPTCHAs
  - Hooks for CAPTCHA solving services

- Building legitimate posting history
  - Posts questions and their answers from different accounts
  - Posts answers to existing questions by stealing answers from other pertinent forums on the Web

- Using anonymizing proxies
  - Discards proxies that expose IP address of posting machine
Advanced Functionalities 2/2

- Spam traffic control
  - Options for speed and success rate
    - Configurable parameters: # of CAPTCHA solving attempts, page size, # of links, # of retrials after timeouts
  - Supports a scheduler
    - Actions taken based on posting finished, timer expiration, number of successful postings

- Reporting
  - Shows success rate for various:
    - TLDs (Top Level Domains)
    - Forum platform software
    - URL patterns
  - Spammers can change strategy based on success rates
Outline

- Introduction
- Overview of Forum Spam Automators
- Primal Functionalities
- Advanced Functionalities
- Traffic Characteristics
- Comparison among Forum Spam Automators
- Conclusion
Traffic Characteristics: HTTP header

- **IE 6 in MS Windows XP**
  - GET or Post \{path\} HTTP/1.1
  - Accept: */*
  - Accept-Language: en-us
  - Accept-Encoding: gzip, deflate
  - User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
  - Host: \{forum host name\}
  - Connection: Keep-Alive
  - Cookie: \{cookie\}

- **XRumer**
  - GET or Post \{path\} HTTP/1.0
  - Accept: */*
  - User-Agent: \{User-Agent string\}
  - Referer: \{visiting URL\}
  - Host: \{forum host name\}
  - Proxy-Connection: Keep-Alive
  - Cookie: \{cookie\}
Traffic Characteristics: Proxy Usage 1/2

- Examination of traffic generated by anonymizing proxies
  - Evaluated 105 public anonymizing proxies
  - Our own client was written in Python
  - Used an Apache Web server
- HTTP headers used
  - Accept, Accept-Language, Accept-Encoding, User-Agent, Host, Connection, Referer
Traffic Characteristics: Proxy Usage 2/2

- **Accept-Encoding** header
  - Removed by 43% of proxies
  - Modified by 9% to ‘text/html, text/plain’
  - *Most modern browsers set it to ‘gzip, deflate’*

- **HTTP headers added by proxies**
  - **Cache-Control** by 47%
  - **Keep-Alive** by 1%
  - **X-Bluecoat-Via** by 3%
  - **X-Forwarded-For** by 1%
## Primal Functions of Forum Spam Automators

<table>
<thead>
<tr>
<th>Functions</th>
<th>XRumer</th>
<th>SEnuke</th>
<th>ScrapeBox</th>
<th>Autopli gg</th>
<th>UWCS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forum platforms</td>
<td>multiple</td>
<td>multiple</td>
<td>3 blog platforms</td>
<td>Pligg</td>
<td>WordPress</td>
</tr>
<tr>
<td>Macro support</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Automatic spam msg. generation</td>
<td>no</td>
<td>yes with additional fee</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Automatic registration</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Automatic posting</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
</tbody>
</table>
# Advanced Functions of Forum Spam Automators

<table>
<thead>
<tr>
<th>Functions</th>
<th>XRumer</th>
<th>Senuke</th>
<th>ScrapeBox</th>
<th>Autopligg</th>
<th>UWCS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learning unknown platform</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>CAPTCHA solving</td>
<td>manual, solving, services</td>
<td>manual, services</td>
<td>services</td>
<td>manual, services</td>
<td>no</td>
</tr>
<tr>
<td>Building a legitimate posting history</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>Reporting</td>
<td>advanced</td>
<td>basic</td>
<td>basic</td>
<td>basic</td>
<td>basic</td>
</tr>
<tr>
<td>Traffic control</td>
<td>advanced</td>
<td>no</td>
<td>basic</td>
<td>no</td>
<td>no</td>
</tr>
</tbody>
</table>
Conclusions

- Forum spam automators
  - Can automate the process of posting forum spam effectively
  - Support various advanced techniques to avoid counter-measurements commonly deployed by forum servers
    - These techniques are sophisticated and evolving
- Future approaches for fundamental forum spam mitigation
  - Heterogeneous posting interface for forum platforms
  - Distinguishing bot behavior from human behavior
    - We are pursuing these approaches in our current work