America's 4 Most Wanted Botnets

Botnet Name	No. of Compromised Computers (US)	Description
Zeus	3.6M	• Key-logging techniques to steal sensitive data • Injects fake HTML forms into online banking login pages to steal user data
Koobface	2.9M	• Spreads via social networking sites with faked messages • Entices user to download codec (malware) to view video
TidServ	1.5M	• Spreads through spam e-mail as attachment. • It uses rootkit techniques to run inside common Windows services or in Windows safe mode.
Trojan.Fakeavalert	1.4M	• Formerly used for spamming • Shifted to downloading other malware

Source: NetworkWorld.com (July 22, 2009)

Spamalytics: An Empirical Analysis of Spam Marketing Conversion

Chris Kanich, Christian Kreibich, Kirill Levchenko, Brandon Enright, Geoffrey Voelker, Vern Paxson, Stefan Savage

ACM Conference on Computer and Communications Security (CCS 2008)

Summarized by Gaspar Modelo-Howard

Abstract

- Measurement study on the “conversion rate” of spam campaigns
  - Probability that an unsolicited email will elicit a “sale”
- Present a methodology using Botnet infiltration
- Analyze two spam campaigns
  - Trojan propagation
  - Online pharmaceutical marketing
- For more than 469M spam emails, authors identified
  - Number that pass thru anti-spam filters
  - Number that elicit use visits to advertised sites (response rate)
  - Number of “sales” and “infections” produced (conversion rate)

Agenda

- Introduction
- Storm Botnet
- Methodology
- Experimental Results
- Conclusions

PURDUE UNIVERSITY

Introduction (1)

- Hard to find person who admits to follow spams "offers", but spam continually clogs mail servers
  - Despite years of energetic deployment of antispam technology
  - Someone must be buying
  - Key questions: how many, how often, how much?
- Problem is limited visibility into basic parameters of spam value proposition
  - Cost to send spam
  - *Probability that email sent will yield a "sale" (conversion rate)*
  - Marginal profit per sale

Introduction (2)

- There are no apparent methods for indirectly measuring spam conversion
  - Best way is to be a spammer
- Authors conducted study by *"sidestepping the obvious legal and ethical problems associated with sending spam"*
  - Ensuring ***neutral actions*** so users never are worse off due to researchers activities
  - ***Reducing harm*** for cases in which user property is at risk
- Method: infiltrate existing spamming botnet, modifying sent spam and directing recipients to authors' websites

Storm Botnet (1)

- P2P botnet that propagates via spam
- Uses two protocols
  - Encrypted version of UDP-based Overnet protocol, based on Kademlia DHT (directory service)
  - Custom TCP-based protocol (C&C)
- Overnet-base protocol messages
  1. Connect
     - During bootstrap phase, node has a initial list of peer
     - Chooses OID pseudo-randomly from 128-bit address space
     - Connects to all peers in list, each available peer returns its own list
     - Node repeats steps for a few rounds
  2. Publicize
     - To let other peers know about its presence
     - Periodically searching for own OID to stay connected and learn about new close-by peers

Storm Botnet (2)

- Overnet-base protocol messages
  3. Search
  4. Publish

  } Export a standard DHT (key.value) pair interface
- DHT keys encode a dynamically changing rendezvous code, to find others on demand
- Bot generates three keys simultaneously: previous, current and next date
  - System clock is set using NTP
  - Keys are used to connect to nodes offering C&C channel
- C&C nodes include their address and port into value and publishes pair to peers close to key

Storm Botnet (3)

- Three classes of Storm nodes
  - Worker bots
  - Proxy bots
  - Master servers
- Very small number of master servers
- If a infected host can be reached externally, becomes proxy
- C&C is pull-based, worker bots request jobs

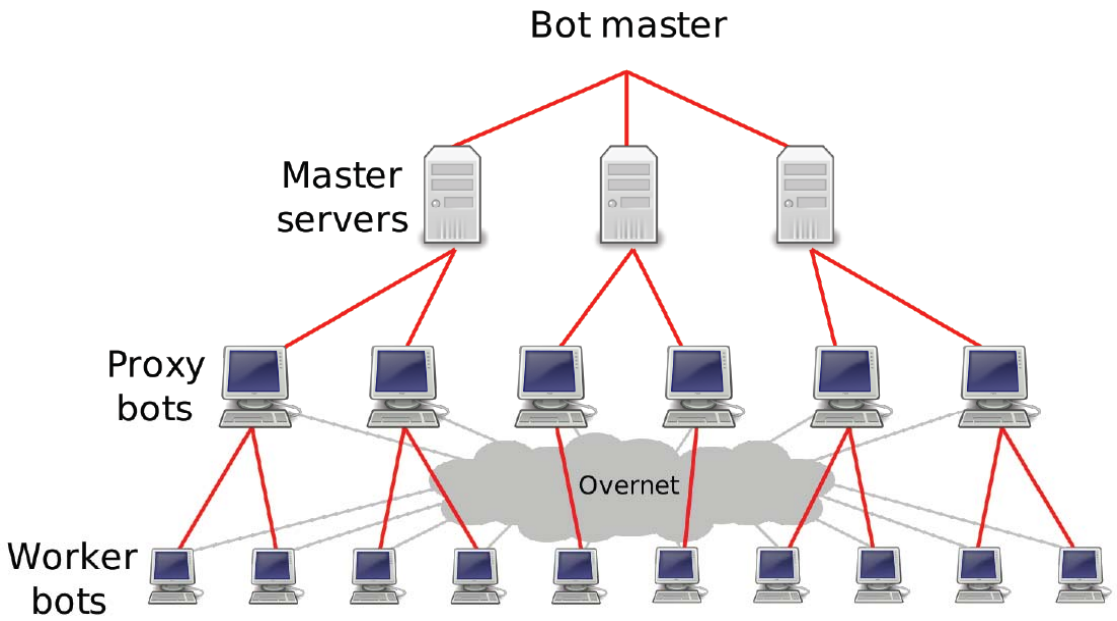


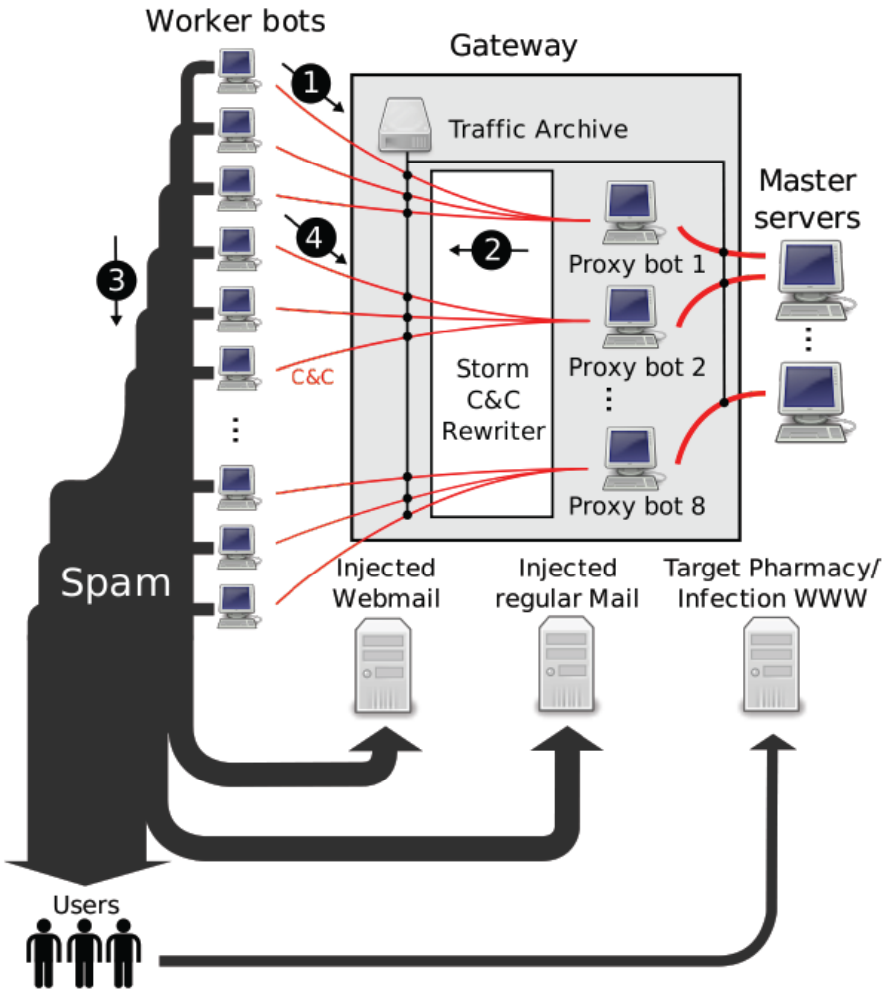
Figure 1: The Storm botnet hierarchy.

Storm Botnet (4)

- Spam engine in detail
  - Bot checks if can reach SMTP server of Web-based mail provider
    - If fails, will remain active but no spam campaigns
  - If successful, finds proxy (using time-varying protocol) and sends update request (via proxy) to master
  - Master responds with spam workload task, which consists of
    - Spam template (use custom macro languages for poly messages)
    - Delivery list of e-mail addresses
    - Set of named "dictionaries"
  - Bot sends unique message for each address to its MX
    - After exhausting list, request two additional spam workloads
    - Then sends a delivery report to proxy (e-mail of recipient if successful)

Methodology (1)

- Based on botnet infiltration
  - Passively observing commands/data and actively changing elements when appropriate
- 8 proxies with gateway
  - Allows for blocking unanticipated behaviors
  - Parsing/rewriting C&C messages to bots



Methodology (2)

- C&C protocol rewriting
  - Click-based network element redirects potential C&C traffic to fixed IP address and port
  - User-space proxy server accepts incoming connections and impersonates the proxy bots
  - Click element injects SOCKS-style destination header into flows to associate connections
- Measuring spam delivery
  - Created collections of test e-mail accounts from Webmail providers, own organization (filtering appliance), and SMTP "sinks" (for control purposes)
  - Rewriter appends these addresses to workloads requests and removes them from success reports
  - E-mail accounts were periodically poll

Methodology (2)

- Measuring click-through and conversion
  - Study focuses on two types of campaigns, self-propagation (rogue postcard sw) and pharmacy site, representing 40% of Storm activity
  - Rewriter replaces any dictionaries with entries only containing URLs to researchers' servers
  - Created two sites to mimic those used in campaigns
    - Pharmacy: no personal/payment information captured
    - Self-propagation: offers benign executable
    - Both sites logged all accesses and activity

Methodology (2)

- Separating users from crawlers
  - Several heuristics created to filter automated or semiautomated processes that visit sites, using blacklist
    - Hosts that access pharmacy site without using unique identifier
    - Hosts that access `robots.txt`
    - Hosts that make malformed requests
    - Hosts that disable javascript and do not load embedded images
    - IP addresses accessing pharmacy site with more than one unique identifier and same `User-Agent` field
    - Hosts that request downloaded postcard executable ten or more times
    - Hosts connecting to rogue IP addresses added to self-propagation dictionary

Methodology (3)

- Screenshot of Pharmaceutical website, operated to measure user click-through and conversion

Experimental Results (1)

- Campaign datasets

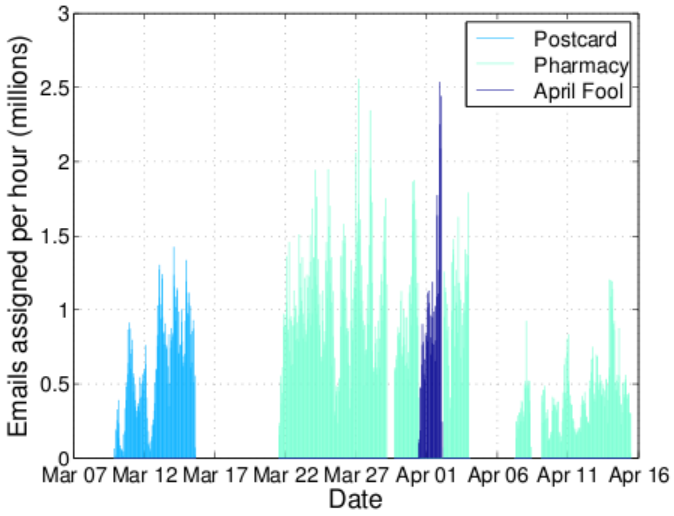


Figure 4: Number of e-mail messages assigned per hour for each campaign.

CAMPAIGN	DATES	WORKERS	E-MAILS
Pharmacy	Mar 21 – Apr 15	31,348	347,590,389
Postcard	Mar 9 – Mar 15	17,639	83,665,479
April Fool	Mar 31 – Apr 2	3,678	38,651,124
		Total	469,906,992

Table 1: Campaigns used in the experiment.

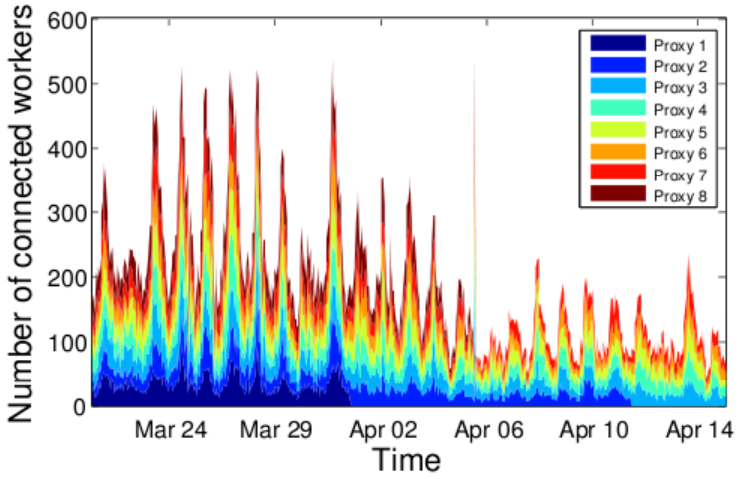


Figure 5: Timeline of proxy bot workload.

DOMAIN	FREQ.
hotmail.com	8.47%
yahoo.com	5.05%
gmail.com	3.17%
aol.com	2.37%
yahoo.co.in	1.13%
sbcglobal.net	0.93%
mail.ru	0.86%
shaw.ca	0.61%
wanadoo.fr	0.61%
msn.com	0.58%
Total	23.79%

Table 2: The 10 most-targeted e-mail address domains and their frequency in the combined lists of targeted addresses over all three campaigns.

Experimental Results (2)

- Spam conversion pipeline

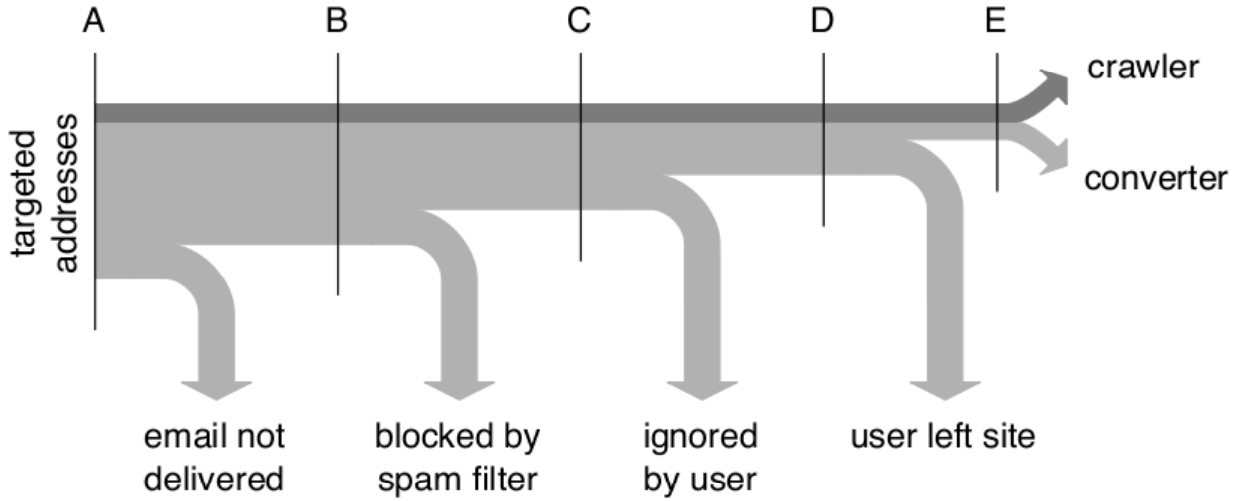


Figure 6: The spam conversion pipeline.

STAGE	PHARMACY		POSTCARD		APRIL FOOL	
A – Spam Targets	347,590,389	100%	83,655,479	100%	40,135,487	100%
B – MTA Delivery (est.)	82,700,000	23.8%	21,100,000	25.2%	10,100,000	25.2%
C – Inbox Delivery	—	—	—	—	—	—
D – User Site Visits	10,522	0.00303%	3,827	0.00457%	2,721	0.00680%
E – User Conversions	28	0.0000081%	316	0.000378%	225	0.000561%

Table 3: Filtering at each stage of the spam conversion pipeline for the self-propagation and pharmacy campaigns. Percentages refer to the conversion rate relative to Stage A.

Experimental Results (3)

- Time to click

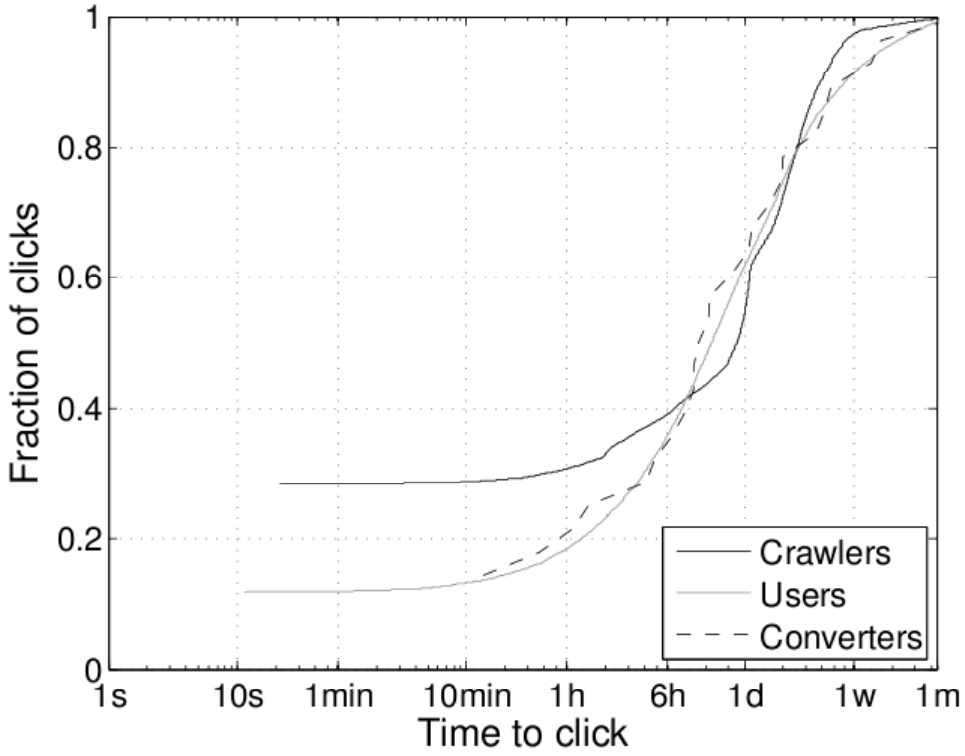


Figure 7: Time-to-click distributions for accesses to the pharmacy site.

Effects of Blacklisting

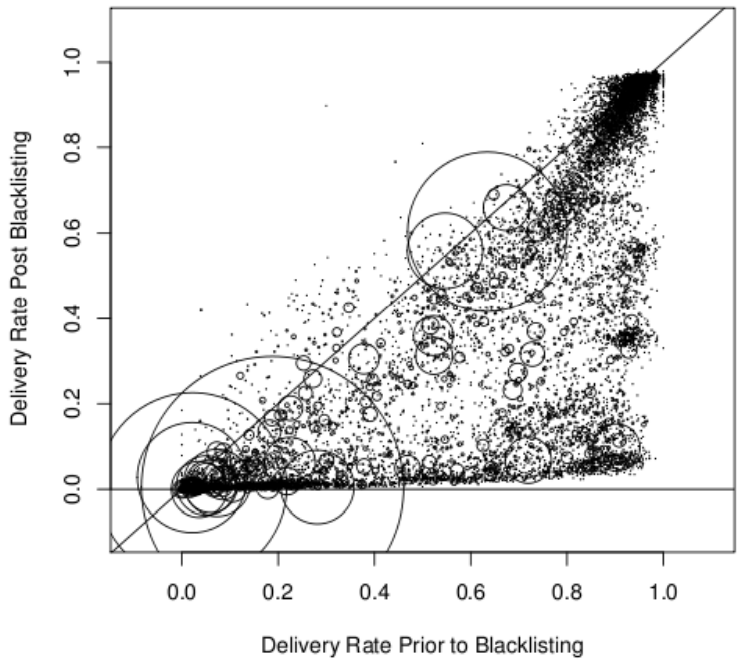
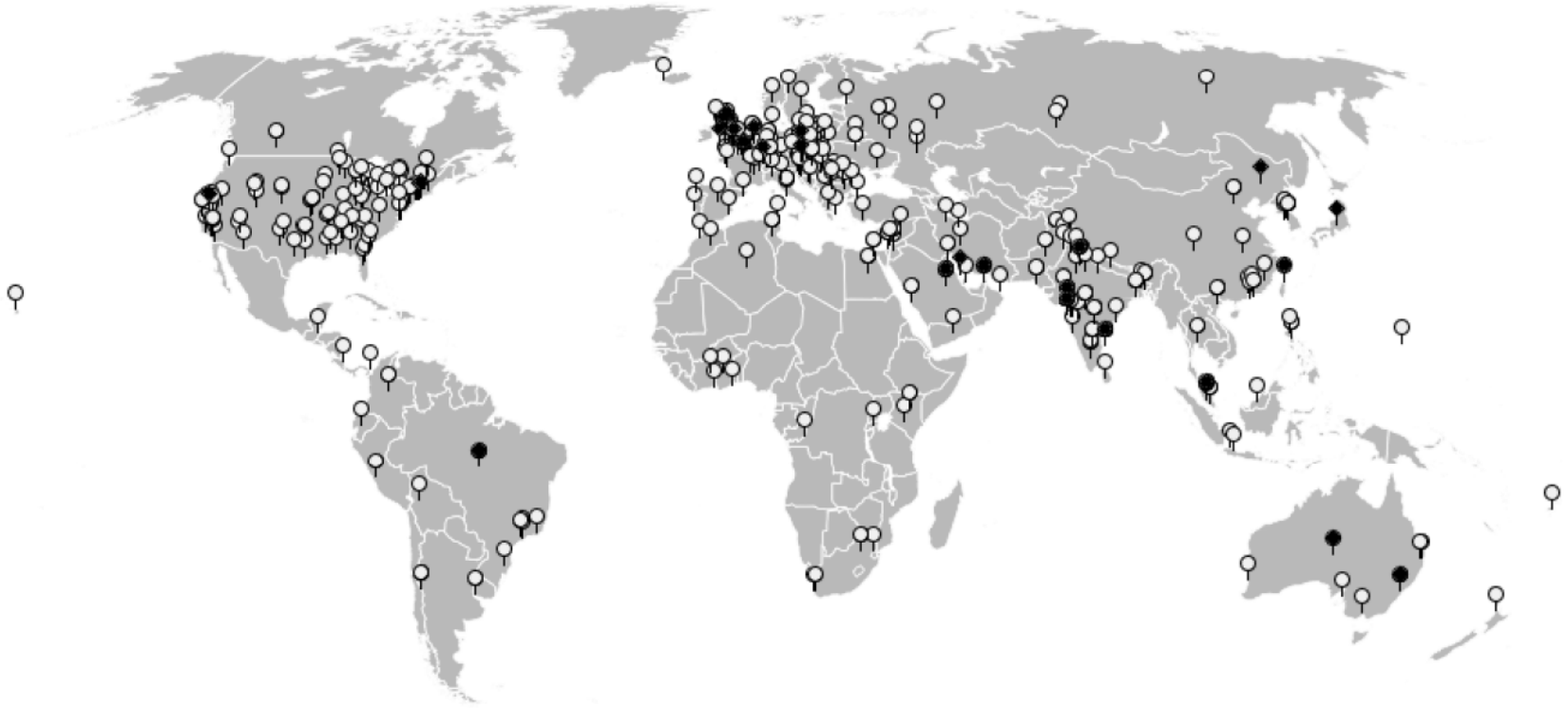


Figure 8: Change in per-domain delivery rates as seen prior to a worker bot appearing in the blacklist (x-axis) vs. after appearing (y-axis). Each circle represents a domain targeted by at least 1,000 analyzable deliveries, with the radius scaled in proportion to the number of delivery attempts.

Conversion Rate Analysis (1)

- Geographic location of “conversion” hosts
  - 541 that executed self-propagation program (gray nodes)
  - 28 that visited purchase page (black nodes)



Conversion Rate Analysis (2)

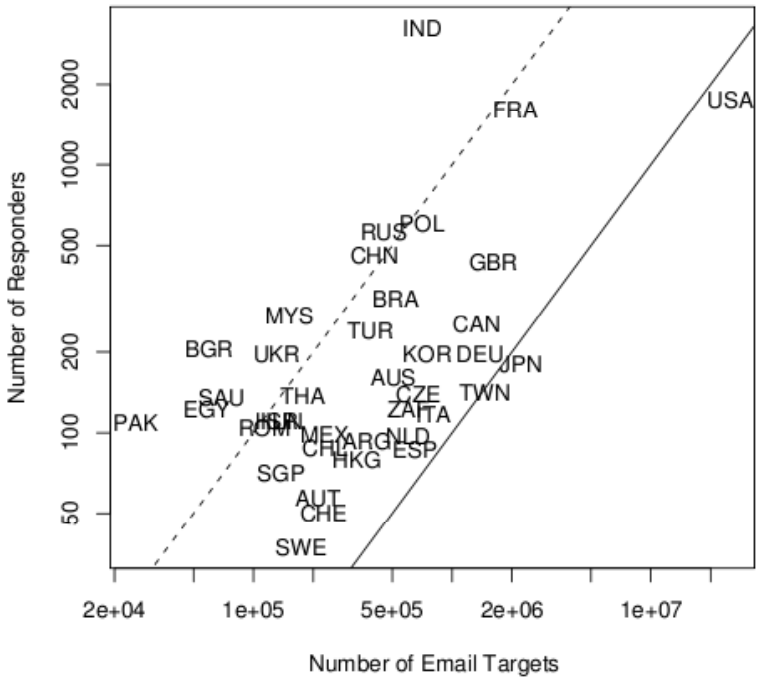


Figure 10: Volume of e-mail targeting (x-axis) vs. responses (y-axis) for the most prominent country-code TLDs. The x and y axes correspond to Stages A and D in the pipeline (Figure 6), respectively.

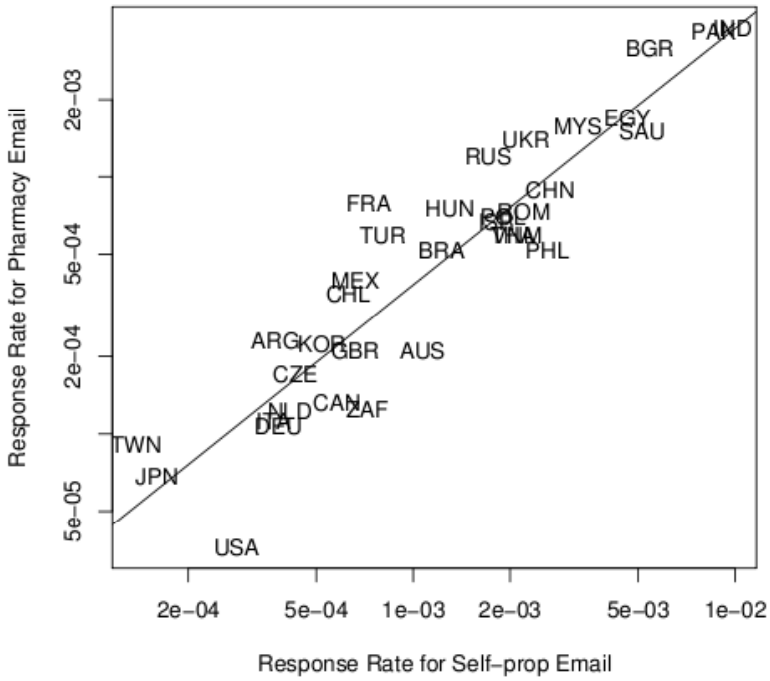


Figure 11: Response rates (stage D in the pipeline) by TLD for executable download (x-axis) vs. pharmacy visits (y-axis).

Conclusions

- Large-scale quantitative study of spam conversion
  - Results represent a single data point and are not necessarily representative of spam as a whole
- Study helps debunk some unscientific claims related to underground economy
- After 26 days, 350 million e-mail messages, only 28 sales resulted
  - Conversation rate: 0.00001% ⟹ revenues of $2731.88
  - Study proxy 1.5% of bots ⟹ $7000 to $9500 per day
- Storm campaigns can produce between 3500 and 8500 new bot per day(estimated)