Corrupted DNS Resolution Paths: The Rise of a Malicious Resolution Authority

David Dagon[1]
Niels Provos[2] Christopher P. Lee[3] Wenke Lee[1]

[1]Georgia Institute of Technology, College of Computing
[2]Google, Inc.
[3]Georgia Institute of Technology, College of Engineering

{dagon@cc., chrislee@, wenke@cc.}gatech.edu
niels@google.com

NDSS 2008

Summary: Resolution Path Corruption

Context: Localized Poisoning

- We measure a growing form of DNS poisoning: resolution path corruption
- Previous: DNS Poisoning against servers
- Today: DNS attacks against stub resolvers
  - stub attacks are known
  - subverting resolution is known
- Contribution:
  - Large-scale measurement
  - We summarize recent trends surrounding malicious open resolvers
  - We measure the extent of DNS path corruption
  - We describe useful measurement techniques
  - We urge further study

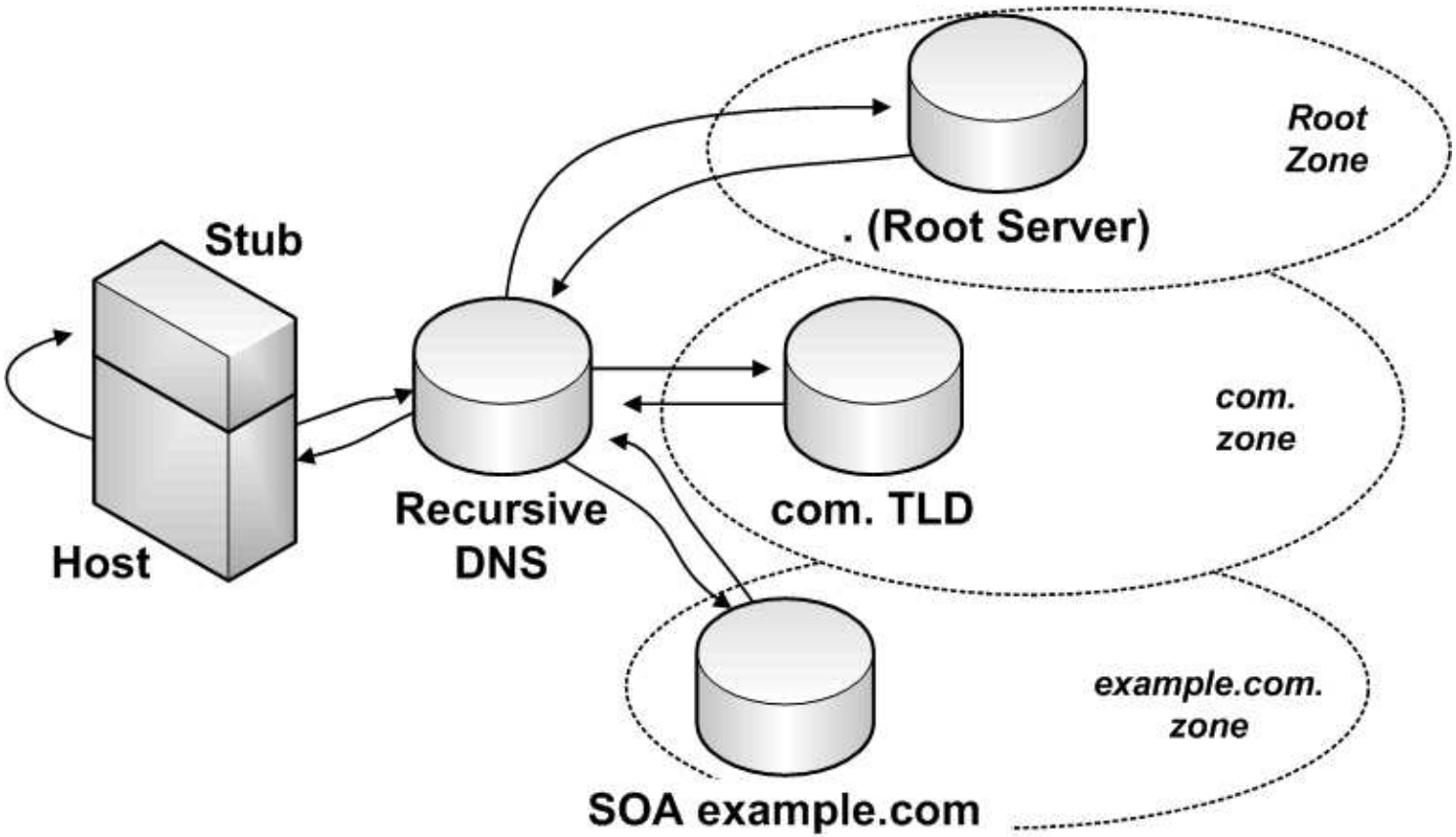
Background

- We have noted a rise in malware that changes default DNS settings
- Many binaries (PE32) point users to malicious DNS servers (e.g., always point to proxies)
- Alarmingly, numerous web pages performed drive-by registry changes
- We decided to investigate

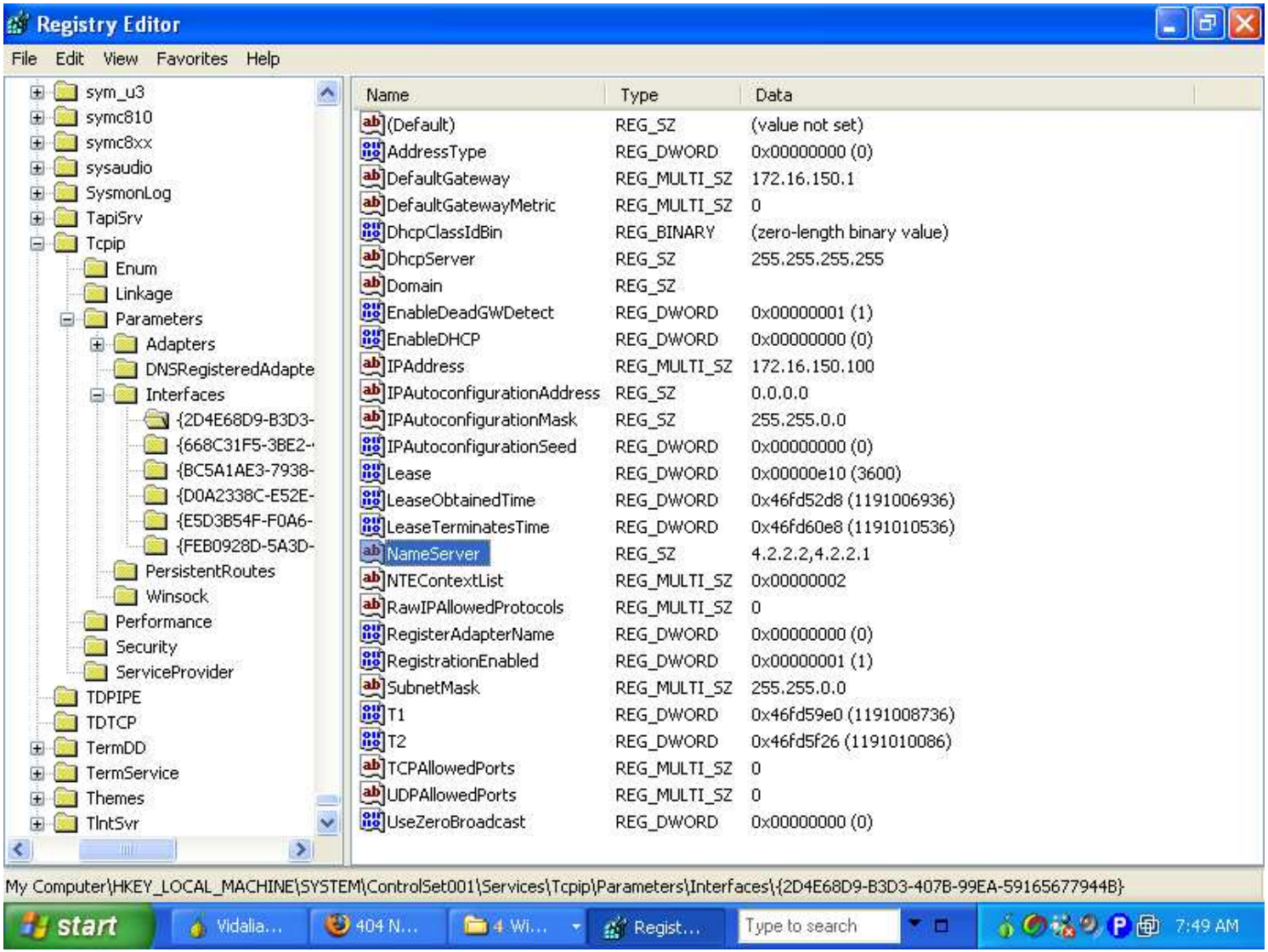
DNS Overview

- Distributed database; tree of labeled nodes
- *Zone*: a clique of nodes, root is *SOA*
- Recursive resolvers surf zone hierarchy to reach SOA or cache point
- Open issue: to what extent must/should a cache respect the wishes of the SOA?
  - Anecdotes of minimal TTL
  - Weak application-caching has spawned a cat-mouse game in dns-pinning/rebinding attacks. (See Jackson, et al. [CCS07])
  - Commercial rewriting of DNS

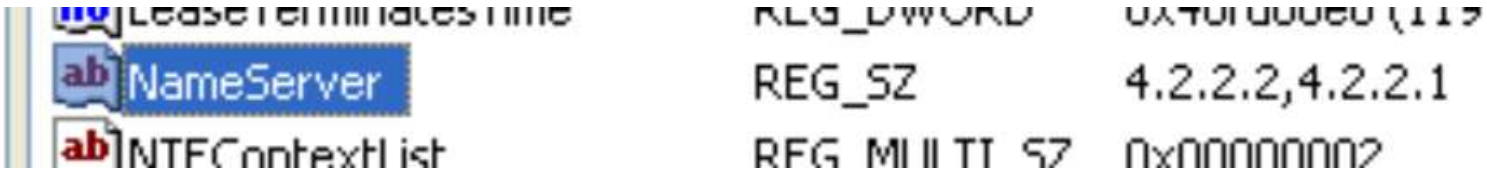
DNS Overview



“DNS Changer” Malware: Normal Setup



“DNS Changer” Malware: Normal Setup

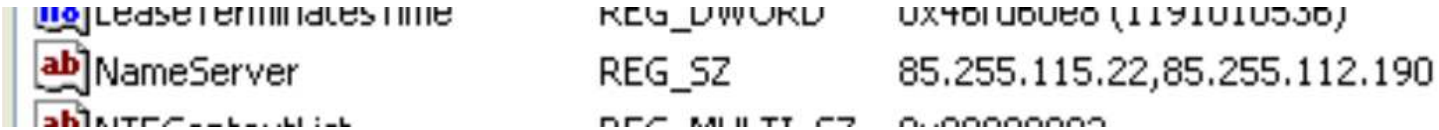


Windows stub resolver users many registry keys, notably
`\\HKLM\SYSTEM\ControlSet001\Services\Tcpip\Parameters\Interfaces\`*`(UID)`*`\NameServer`

“DNS Changer” Malware

- Malware is introducted through the usual vectors (e.g., e-mail spam, web link spam, social engineering)
- Anecdote: Site distributing DNS-changing `zcodec` trojan was top 15,000 page on Internet (3 Yr. Alexa Ave.)
- See also: recent zlob outbreak

“DNS Changer” Malware: Result



- Sometimes, additional malware dropped (banner/adware)
- Beyond that, the only evidence is the DNS change.
- Consider the challenge this presents to anti-virus detection
  - How does an AV know a DNS server is malicious?

Analysis Challenge

Is this malicious or misconfigured?

```
; «» DiG 9.3.4-P1 «» @ns5.namerich.cn.  +trace
any zksw.com.
; (1 server found)
;; global options:  printcmd
.  86400 IN NS ns5.namerich.cn.
.  86400 IN NS ns6.namerich.cn.
; Received 94 bytes from
220.194.59.57#53(220.194.59.57) in 600 ms
zksw.com.  300 IN A 210.72.13.14
com.  86400 IN NS ns6.namerich.cn.
com.  86400 IN NS ns5.namerich.cn.
;; Received 121 bytes from
220.194.59.57#53(ns5.namerich.cn) in 600 ms
```

Likely Misconfiguration

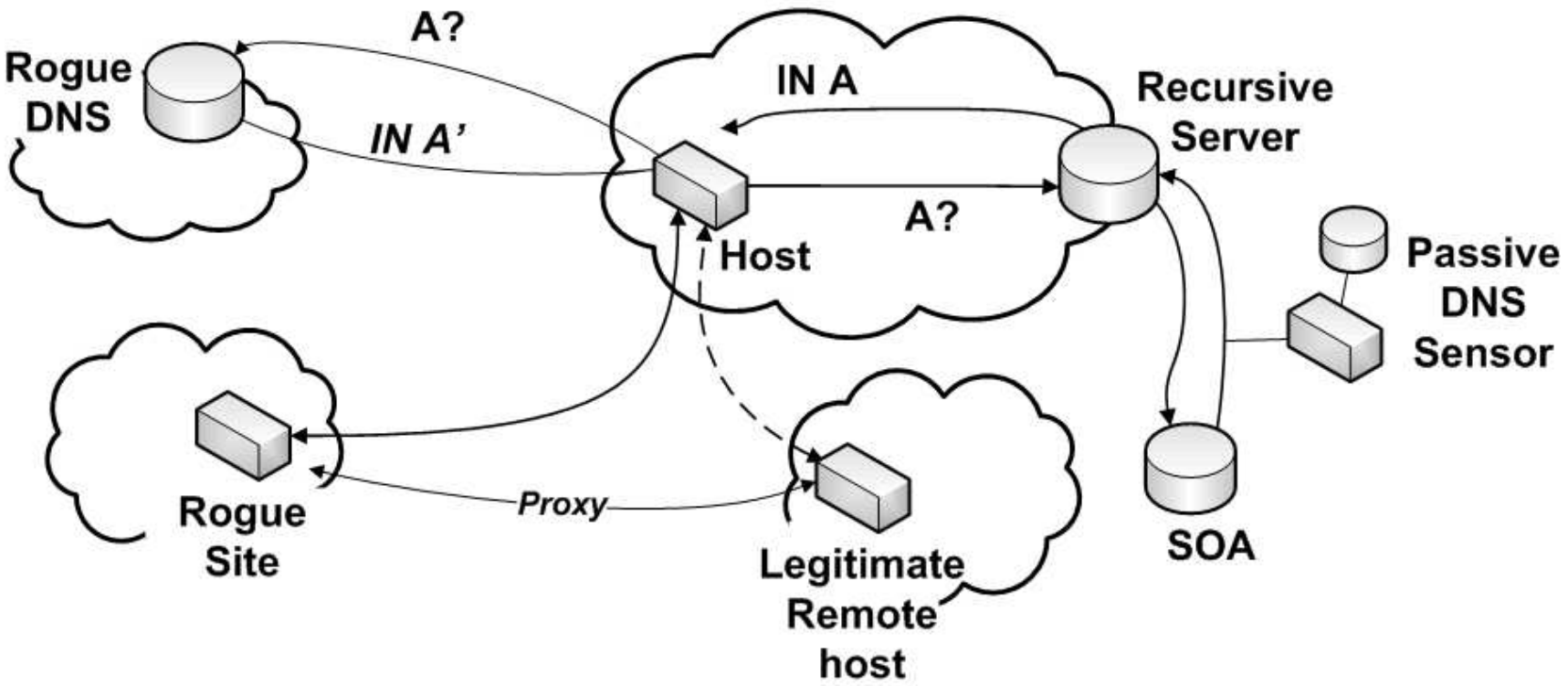
```
$ORIGIN com.
@          IN   SOA   ns6.namerich.cn.  (
           2006072701   ; serial poisoning since
2006
           7200       ; refresh
           3600       ; retry
           3600000      ; expiry
           3600 )       ; minimum
           IN    NS    ns5.namerich.cn.
           IN   NS   ns6.namerich.cn.
zksw        IN   A   210.72.13.14
```

Conclusion: using just IP addresses, it's hard to determine infection/non-infection

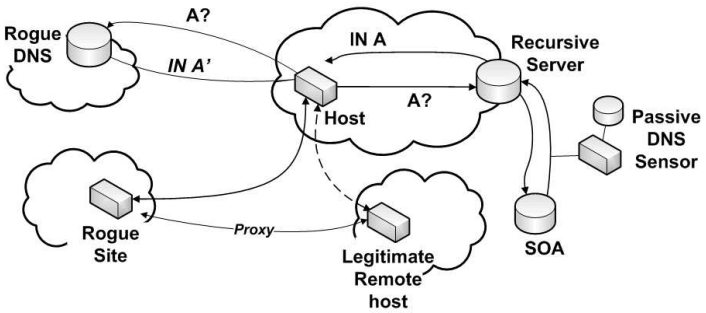
“DNS Changer” Malware: Autopsy

- In this case, it’s likely misconfiguration
- Key: it can be difficult to use IP addresses alone to detect “bad” DNS settings
- Implications for malware that alters DNS settings:
  - Complete control over resolution
  - Difficult to detect
  - Trivial proxying/injection or replacement of content
- Part of larger issue: pharming
  - See also DNS-related talks of John Kristoff
  - See Univ. Indiana/Symantec Study (Alex Tsow, “Phishing with Consumer Electronics: Malicious Home Routers”)

“DNS Changer” Malware: The Big Picture



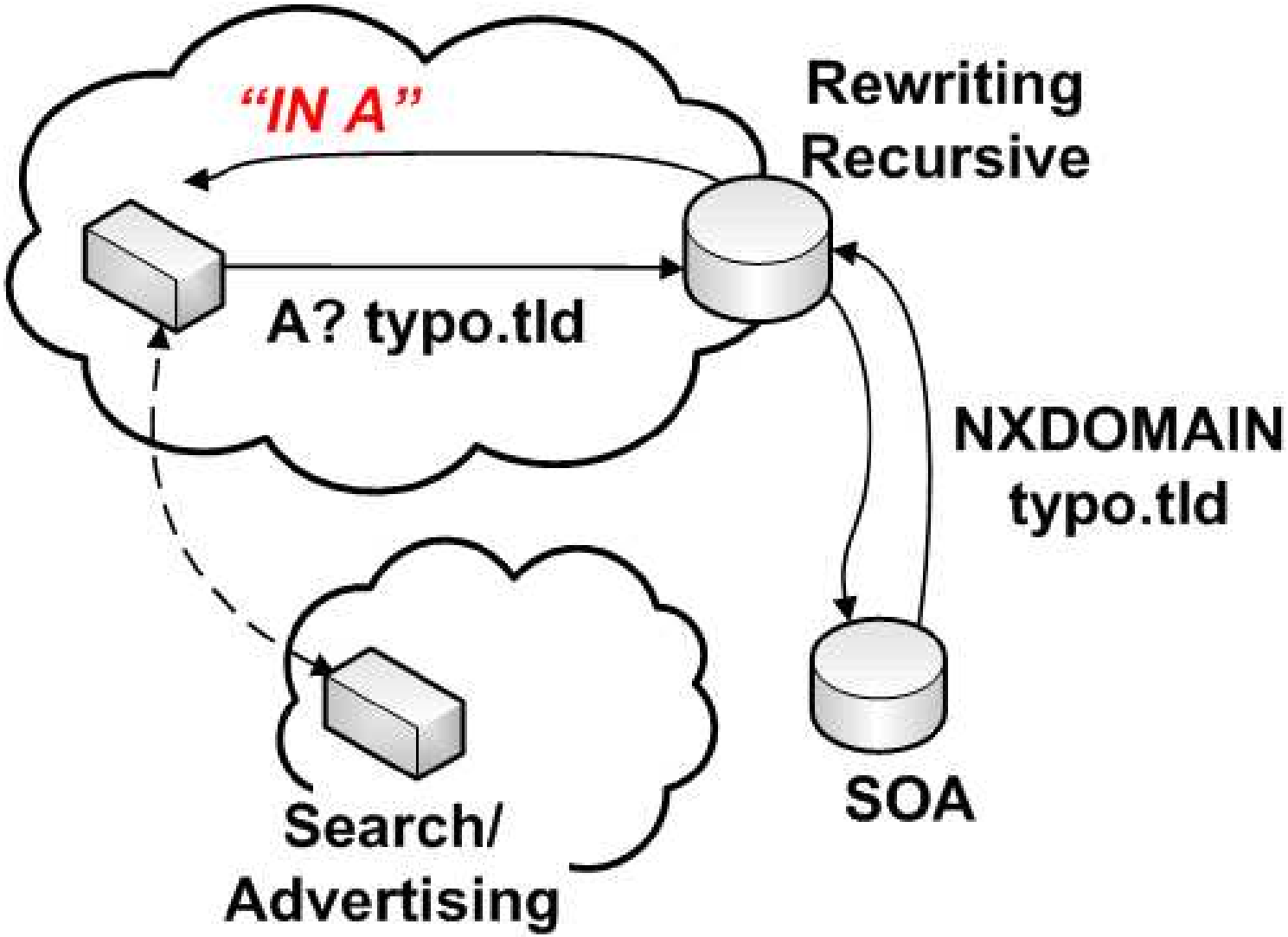
“DNS Changer” Malware: The Big Picture



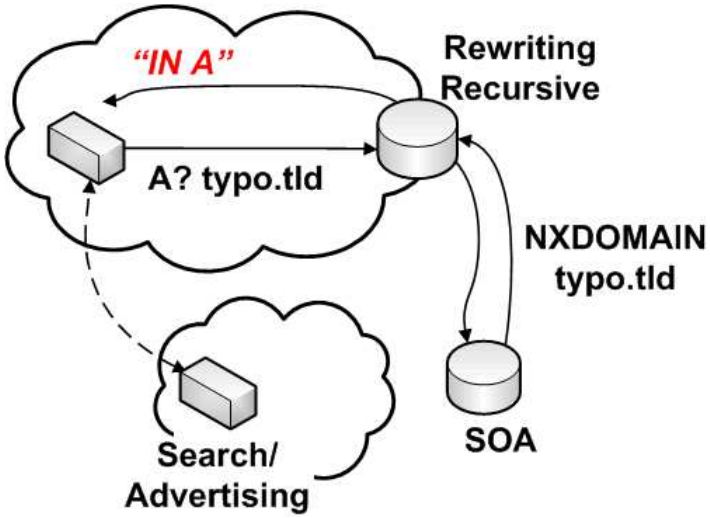
- Malware trivially changes resolution settings
- Rogue DNS server selectively provides malicious answers
- Web servers proxy connections/logins (even without complete MIM)
- Farms of “rogue” DNS servers spotted. (See also Trend Micro’s blog[1] entries).

[1] `http://blog.trendmicro.com/rogue-domain-name-system-servers-5breposted5d/`

“DNS Rewriting”: The Background



“DNS Rewriting” The Background

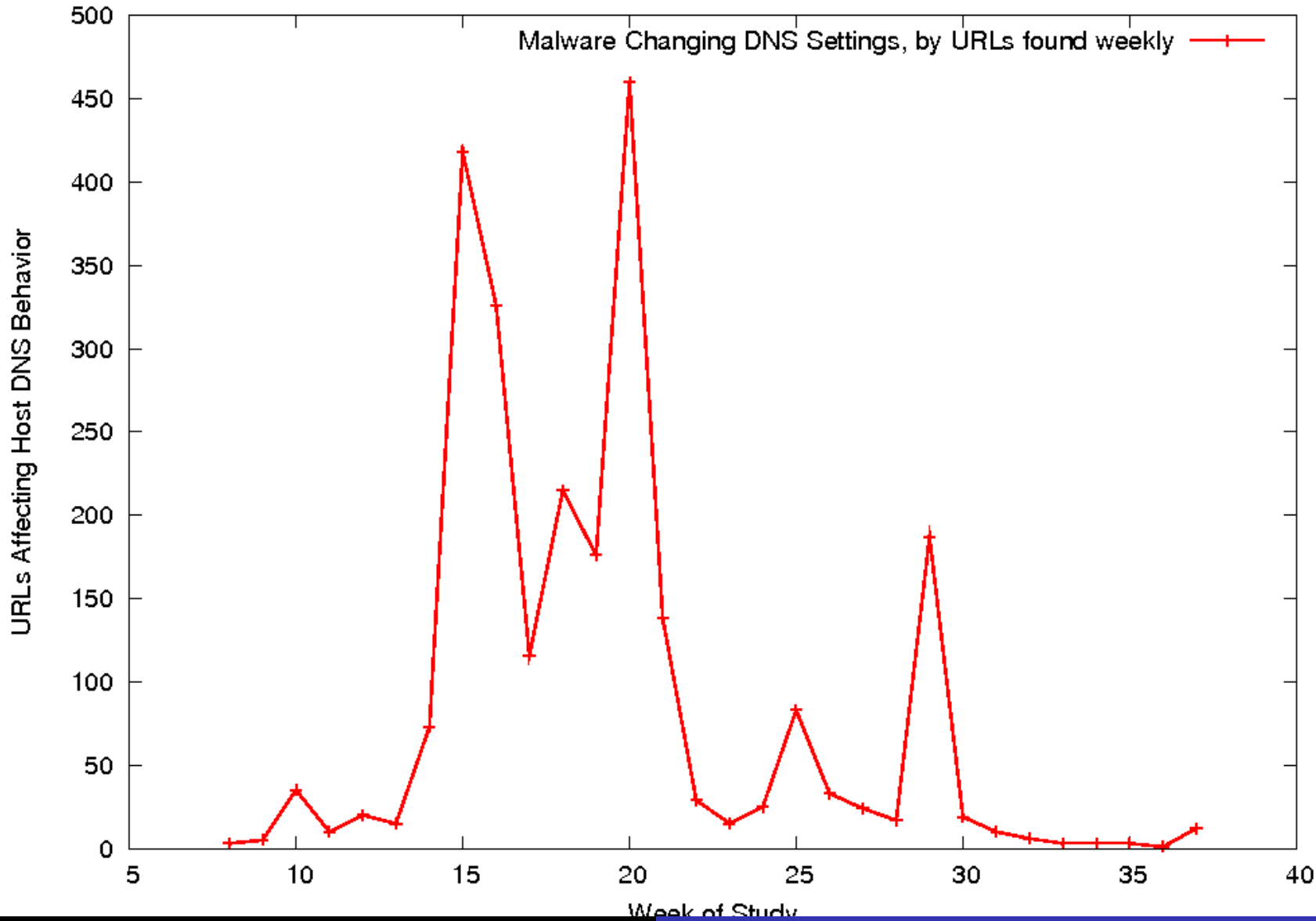


- ISPs often rewrite DNS packets, e.g., for `NXDOMAIN`
- So-called error path correction
- No RFC prohibits this; not considered “spec” by many
- Distinguished from malicious behavior: *consent* of end host
- Key idea: DNS is a *consensus reality*.

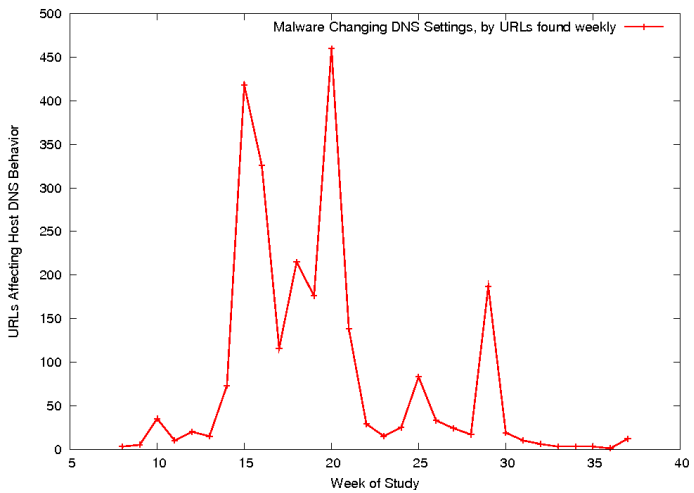
“DNS Changer”: Prevalence of Malware

- How extensive is this problem? How much malware changes DNS settings?
- Study using `malfease.oarci.net`
- $\sim$ 200K samples gathered
- $\sim$ a dozen changed DNS settings
- What else could be altering DNS resolution path?
- We need a much larger study sample

“DNS Changer” Drive-By Web Attacks



“DNS Changer” Drive-By Web Attacks

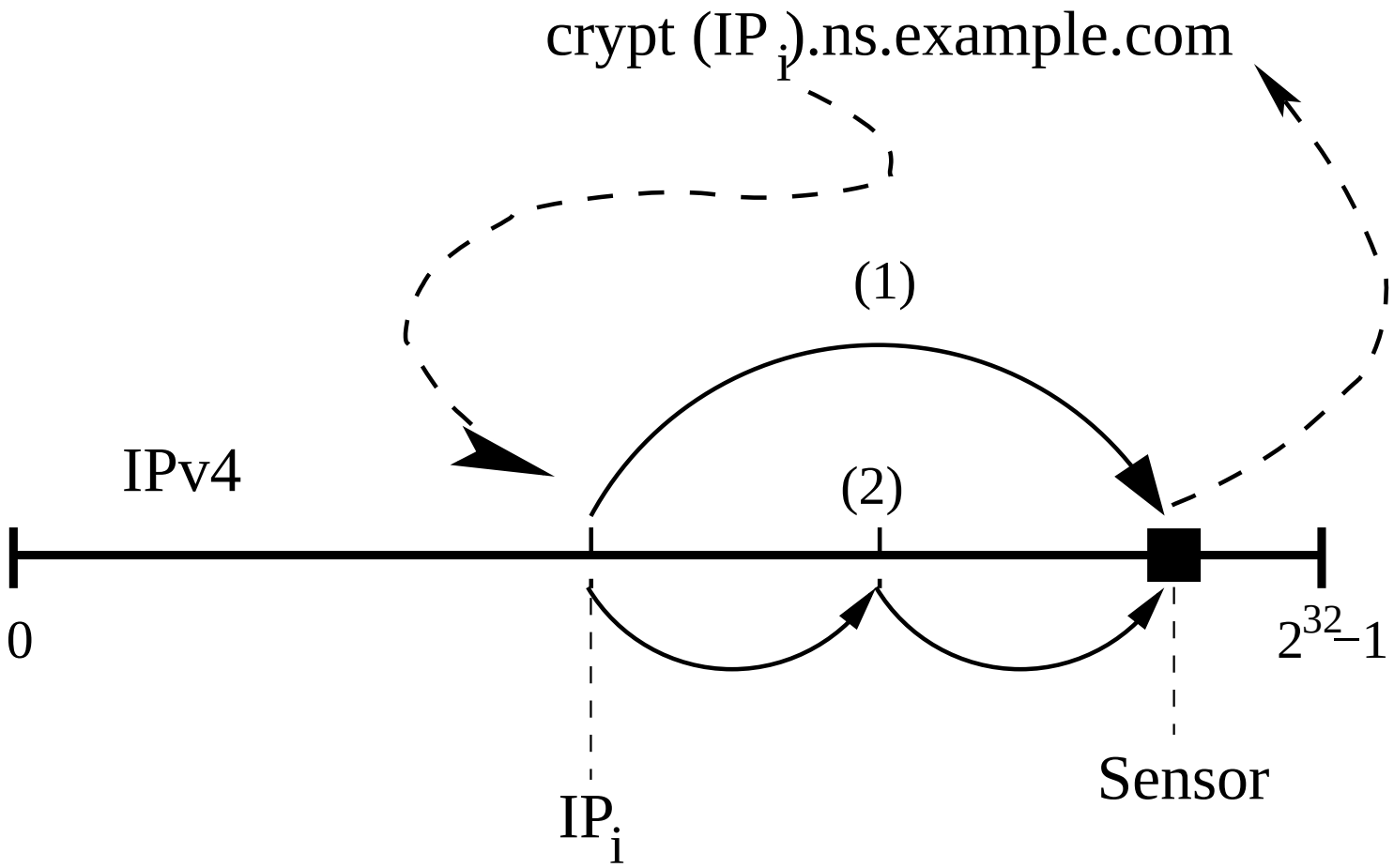


- Google checked the previous months of crawls
- Hundreds of web pages per week were discovered that change DNS settings (2,100 pages over 600 domains, pointing to 75 unique DNS servers).
- No insight as to age of page; given the source, one suspects the pages were discovered early.
- Note: Google offers a related domain reputation API.

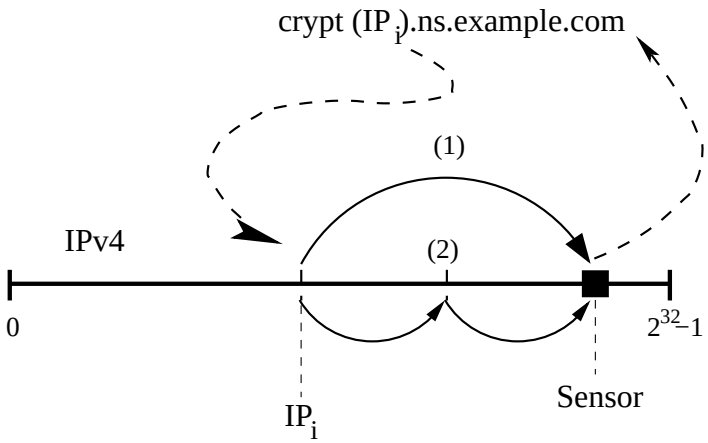
Sourcing Resolution Path Corruption

- We verified this attack using passive DNS (and full captures) at campus border
- Who is behind this?
- Note: registry key changes are *trivial*
  - One merely has to run a rogue DNS server
  - ... or become an affiliate of such a rogue server
- Beyond the anecdotal rogue DNS servers, we know:
  - These attackers use IPv4;
  - These run open resolvers (by necessity, absent complicated victim ACLs)
- We decided to “round up the usual suspects” and question them in the lab.
  - We first needed to locate open resolvers...

Study Methodology



Study Methodology



- Unique label queried to all IPv4
- S0A wildcard for parent zone
- Script used to return srcIP of requestor
- Logging at NS yields open recursive and recursive forwarding hosts

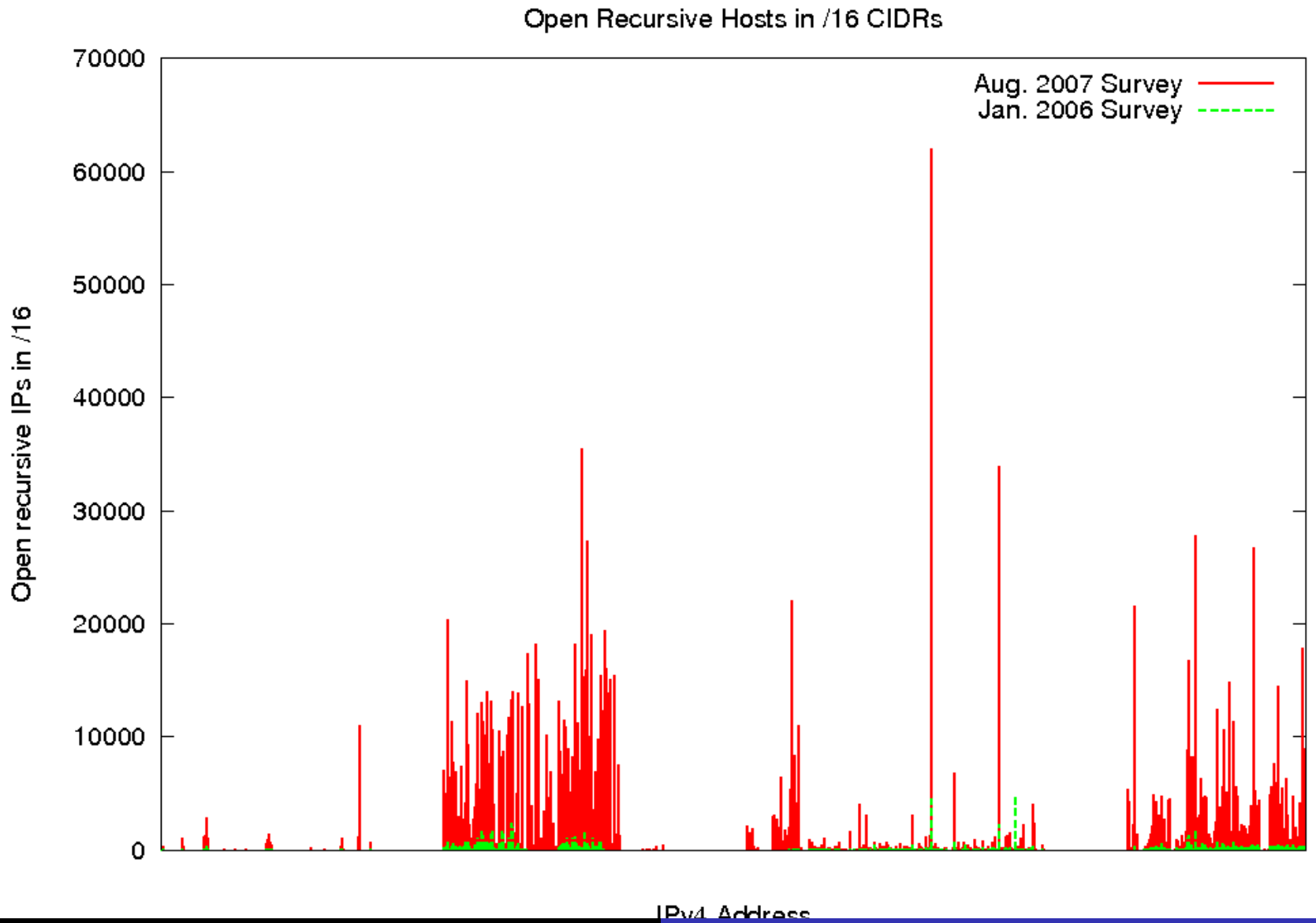
Design Goals for Survey

- Policy, policy, policy
  - Exclude bogons, mil, gov, etc.
  - Follow RFC 1262's advice
    - The PTR gave clues ("`dnsstudy1`")
    - Web page provided means of self-exclusion
    - Responsive `abuse@` group created
    - Apologies to those with noisy IDS gear
- Child label considerations
  - Save state (stop, restart)
  - Avoid caching (unique labels)
  - Trivially reversible (avoid SELECT)
- Other strategies:
  - Embed srcIP in RR
  - Lamport hash of IPs (cf. SSH Scan tools)

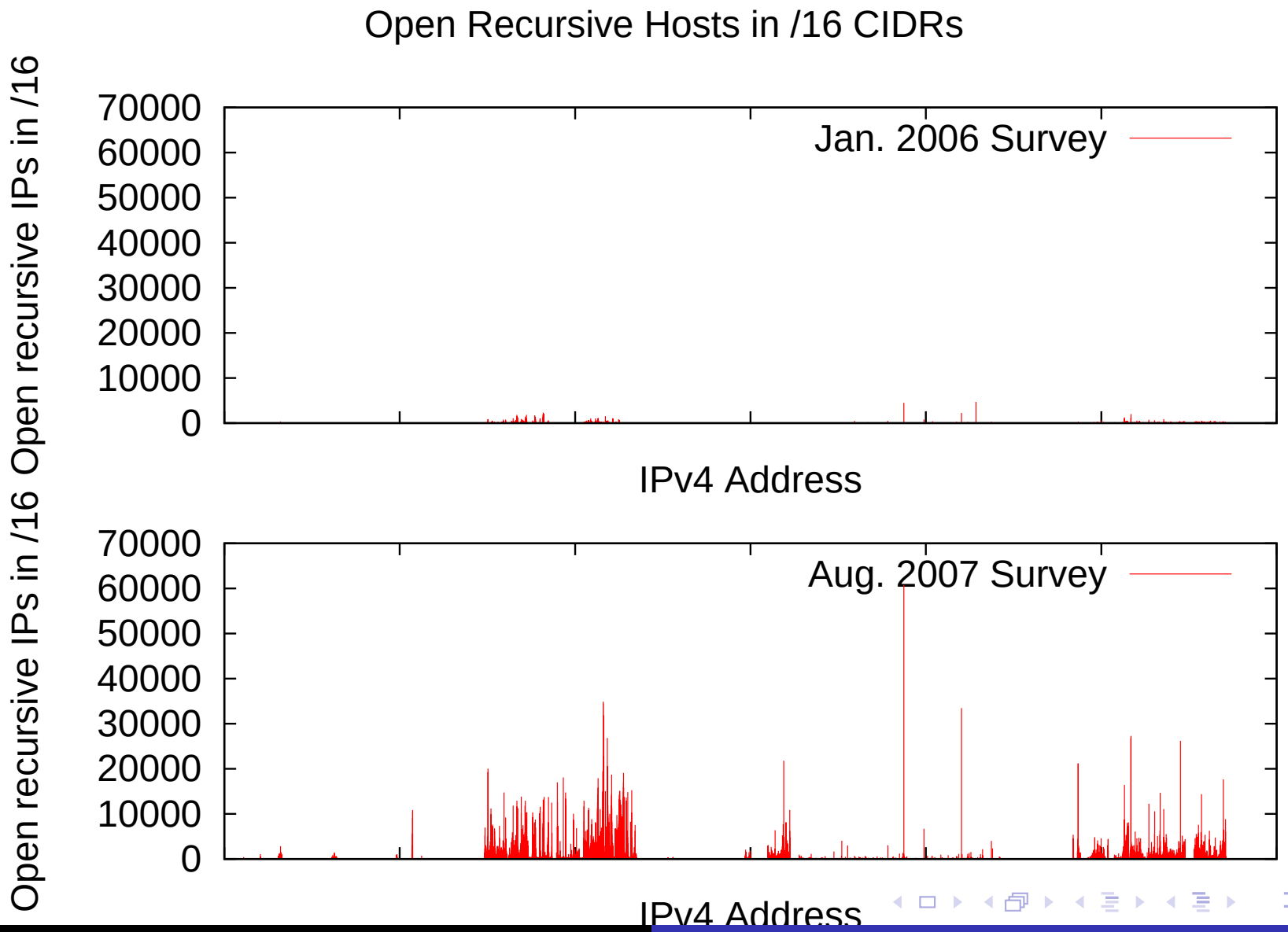
Methodology (cont'd)

- Phase1
  - If response given...
  - Exclude authority open resolvers
  - `fpdns` taken of answering host
  - Perform http request of host
- Phase2
  - Pick 600K open resolvers
  - Ask them repeatedly to resolve phishable domains
  - Note which ones gave incorrect answers
  - If "incorrect", http request to the answered IP

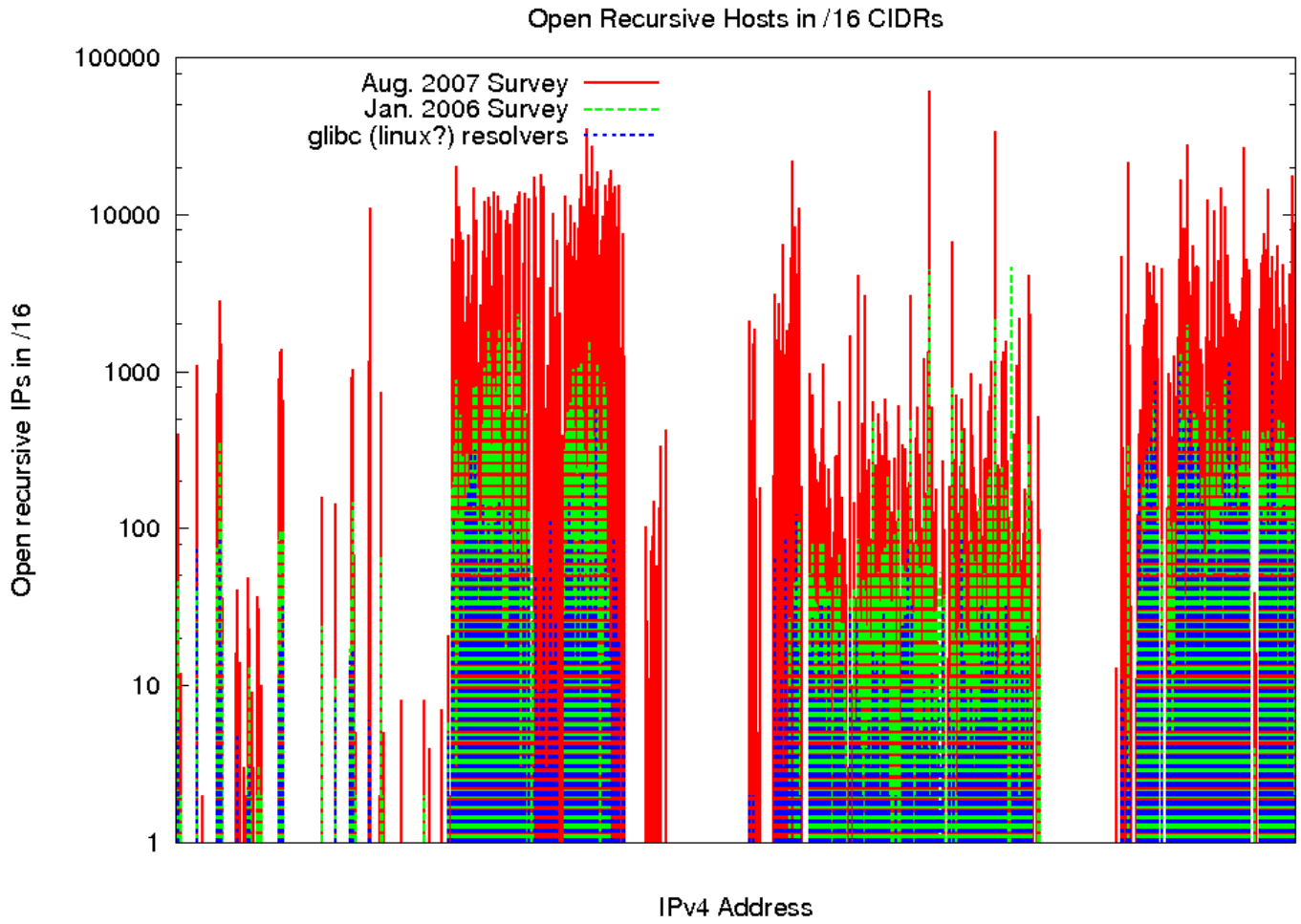
Open Recursion: Comparison of /16s, in IPv4



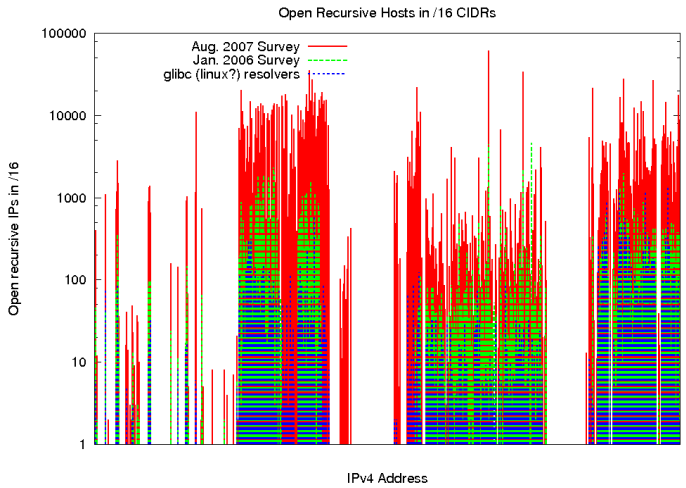
Open Recursion: Comparison of /16s, in IPv4



Open Recursion: Putative GNU libc /16s



Open Recursion: Putative GNU libc /16s



- gnu libc logic of AAAA? $\longrightarrow$ A? queries.
- Other heuristics: Windows DNS servers answered authoritatively for queries for `1.in-addr.arpa`,
- Needed item: update fpdns (2005)
- Other “harmless” explanations considered, discarded

Analysis: Open Resolvers

- Two sweeps of IPv4:
- Aug 2007, 10,427,000 open recursive
- Sep 2007, 10,573,000 open recursives
- Union: 17,365,000 open recursives over 2 weeks
- Intersection: 3,634,000 in common
  - Some packet loss perhaps
  - However, *union count* points to mass migration of 7M hosts
- Multiple subsequent full sweeps of IPv4 put numbers at 16-17M, depending on time of day(!)

Analysis: What DNS Server is Running?

- HTTP server string fetched from open recursive hosts
  - $\sim$ 20% RomPager, Nucleus, misc. known devices
  - $\sim$ 80% No answer
- Thus, designed study groups:
  - Randomly selected open recursive resolvers
  - Intersection of open recursives and visitors to Google's authority server
  - Intersection of open recrusives and Storm victims

Analysis: “DNS Liars”

- Methodology:
  - selected 200K random open recs, 200K open recs contacting Google authority servers, 200K overlap storm
  - Repeatedly queried for “phishable”; 15 min window; 220M probes total over 4 days
  - Diurnal pattern noted (unusual for DNS servers)
  - Approx. 310K-330K resolvers answer; 460K out of 600K total answered
    - Recall migration among 10M open resolvers, noted above
- 2.4% “lied” (extrapolates to 291,500K hosts)
- 0.4% were malicious (extraploates to 68K hosts; 36K measured so far in subsequent full IPv4 sweeps)
- Created database of “proxied” webpages
  - Porn, advertising, and proxied pages(!)
  - $\sim$ 20% proxied/rewrote google.com (demo)
  - $\sim$ 11% proxied a chinese search page
  - $\sim$ 26% proxied a comcast user login

Conclusion

- DNS is undergoing a monetization makeover
  - Commercial “error-path correction” at recursive level
  - Malicious alteration of resolution path
  - The distinction: consent
- Numerous virus and hundreds of web pages *automatically* change user DNS settings
- Difficult to detect
  - Rogue DNS servers *sometimes* lie
  - The IP of the NS *alone* is (generally) insufficient)
- The security community needs to propose solutions:
  - DNSSEC, DLV, DNS reputation, blocking, recovery, measurement
  - Hopefully, we can avoid balkanization of networks, and loss of e2e for DNS resolution.

Probe Strategies: Ongoing Mapping

- Ongoing work:
- About every week, rescan IPv4
- About every hour, rescan “hot CIDRs”
- Poll to known “old” DNS servers for early poison detection
- Data available via OARC distribution model

Acknowledgements

Georgia Tech Campus
(Cross Sectional View)

We are grateful for the assistance of:

- *IPv{4,6}:* Nicholas Bourbaki
- *Google:* David Presotto and Ankur Jain
- *ISC:* Paul Vixie
- *OpenDNS:* David Ulevtich
- *InfoBlox:* Cricket Liu
- *Gatech CS:* Robert Edmonds
- *M5 Hosting:* Peter Honeyman
- *Gatech OIT:* The entire `abuse@` staff!