Advances in PassiveDNS Replication

FIRST 24, Malta
19 June 2012

Architecture: Robert Edmonds
Presented by: Eric Ziegast
Internet Systems Consortium, Inc.
Agenda

- Review of PassiveDNS Replication
  - How it works, Why it's useful, History, Evolution
- Sensors
  - Evolution, Hardening, Privacy, Software, Relaying
- Data processing
  - Scalable multi-stage processing and data flow
  - Deduplication, Filtering, Verification
- Database
  - Lessons learned
  - Evolution
- Access
- Community / Goals
How it works (1\textsuperscript{st} client)

1. client 1 queries the caching server with `www.isc.org? (rd=1)`.
2. The caching server resolves the query to `www.isc.org`.
3. The org namespace is queried and the result is cached.
4. Client 2 can then access `www.isc.org` without querying the caching server again.
How it works (query/response)

client 1

client 2

root ns

www.isc.org? (rd=1)
A 149.20.64.42

resolving ns

www.isc.org? (rd=0)

org ns

www.isc.org? (rd=0)

isc.org ns

www.isc.org? (rd=0)

isc.org ns

A 149.20.64.42
How it works (2\textsuperscript{nd} client)

client 1

client 2

www.isc.org? (rd=1)

A 149.20.64.42

caching server

resolving ns

isc.org ns

org ns

root ns
History

- Florian Weimer started in 2004
- Public efforts (RUS-CERT, BFK, DN Sparse, CertEE, CIRCL, CERT.AT)
  - One tool to use them all (Chris Lee):
    http://code.google.com/p/passive-dns-query-tool/
- Private efforts (TeamCymru?, AV Vendors, NOTOS)
- Most use PCAP-based tools (like tcpdump or dnscap) to capture packets, extract data, add to SQL data base, develop query tool (whois)
Evolution

- Vixie started in 2007, Edmonds in 2008
- Saw challenges in existing tools
  - dnscap -> ncaptool -> nmsgtool
- Goals:
  - Making it easier to deploy
  - High volume replication and processing
  - Real-time by-products
  - Optimizing data storage and access technologies
Sensors

DNSDB
Most focused on UDP responses

We did too at first...  
... but that's not good enough.
PassiveDNS Hardening

Learn more: (Edmonds @ DEFCON18): http://bitly.com/IAJHVZ
Privacy

Personally Identifiable Information

High volume

www.isc.org? (rd=1)
A 149.20.64.42

Useful for finding who is affected by badness (like infected clients)

Generally*
free of PII

Low volume

www.isc.org? (rd=0)
org ns

Useful for mapping badness and detecting changes

isc.org ns

A 149.20.64.42

Generally*
free of PII

www.isc.org? (rd=0)

A 149.20.64.42

Useful for finding who is affected by badness (like infected clients)
Privacy

- Filtering – sensor tool can filter out local domains or zero out nameserver
- Aggregation – How many users are behind a nameserver? (one? 1,000? 100,000? more?)
- Aggregation – Our processing framework strips out sensor nameserver information
- Aggregation – Sensor data from multiple operators are mixed together
- Concern?: Admins putting PII data into query strings or responses
- Counter: DNS information is “published”
Sensor (ns)

Placement of sensor software (on nameserver)

- Software runs on nameserver
  - Minimal cpu usage compared to nameserver
  - Tunable maximum memory usage for hash cache
    (prefer 256MB-512MB)

- Configuration uses upstream address for BPF filters.
  - What IP address does nameserver use when querying auth servers?
  - What interface do queries/responses leave/return?
    (eg: “eth0”)

- No forwarders please
  - Want auth answers only without TTL changes

- Prefer many clients per recursive nameserver (1000+)
  to help maintain PII privacy
Sensor (tap)

Placement of sensor software (network-wide tap)

Switch configured to mirror interfaces to monitoring server or use tap on router downlinks (eg: IDS configuration).

Software runs on monitoring server.
- No CPU or memory footprint on nameservers.
- General “catch-all” of pDNS for entire network

Uses promiscuous mode (eg: “eth0+”) for interface.

No addresses to configure.
What are PII concerns for individuals running resolvers?
Sensor (span)

Placement of sensor software (port mirroring)

- Switch configured to mirror interfaces to monitoring server.
- Software runs on monitoring server.
  - No CPU or memory footprint on nameservers.
  - Good for HA or high-volume environments.
- Uses promiscuous mode (e.g., “eth0+”) for interface.
- What IP subnet or list of addresses do nameservers use for upstream queries?
Sensor Software

- Open source
- Binaries (Linux packages):
- Scripts (FreeBSD, other):
- Installs nmsgtool, wrapsrv, shell scripts
- Edit config file based on placement
- Captures ISC:dnsqr data to file
- Robust rsync upload
Why it's useful

- Robust criminal infrastructure uses DNS
- See abuse in real time
- Criminals will keep (re)using infrastructure until it's taken away
- Reverse indexing -> associations
- DNS History – track changes
**Guilt by association**

<table>
<thead>
<tr>
<th>Domain</th>
<th>First Seen</th>
<th>Last Seen</th>
<th>IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>bailwick</td>
<td>2010-11-15 02:47:01 -0000</td>
<td>2010-11-26 02:07:10 -0000</td>
<td>190.86.101.171</td>
</tr>
</tbody>
</table>

Criminal Domain Names found via the bad A Record

Criminal Domain Names found via the bad Name Server
Common resources

Found 4 RRsets in 0.00 seconds.

<table>
<thead>
<tr>
<th>bailiwick</th>
<th>count</th>
<th>first seen</th>
<th>last seen</th>
<th>first seen in zone file</th>
<th>last seen in zone file</th>
</tr>
</thead>
<tbody>
<tr>
<td>com.</td>
<td>50182</td>
<td>2010-06-24</td>
<td>2011-09-05</td>
<td>2010-08-01</td>
<td>2011-09-04</td>
</tr>
</tbody>
</table>

betfair.com. NS udns1.ultradns.net.
betfair.com. NS udns2.ultradns.net.

Search mode: RRset Rdata
Record type: NS
Record data: ns1.yumurtakabugu.com
Input mode: Name IP or network Raw hex

Search
Reset

Found 20 RRs in 0.02 seconds.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>cryjester.net.</td>
<td>NS</td>
<td>ns1.yumurtakabugu.com.</td>
</tr>
<tr>
<td>dell.co.kr.</td>
<td>NS</td>
<td>ns1.yumurtakabugu.com.</td>
</tr>
<tr>
<td>freegary.co.uk.</td>
<td>NS</td>
<td>ns1.yumurtakabugu.com.</td>
</tr>
<tr>
<td>freegary.org.uk.</td>
<td>NS</td>
<td>ns1.yumurtakabugu.com.</td>
</tr>
<tr>
<td>hable.co.kr.</td>
<td>NS</td>
<td>ns1.yumurtakabugu.com.</td>
</tr>
<tr>
<td>telegraph.co.uk.</td>
<td>NS</td>
<td>ns1.yumurtakabugu.com.</td>
</tr>
<tr>
<td>theregister.co.uk.</td>
<td>NS</td>
<td>ns1.yumurtakabugu.com.</td>
</tr>
</tbody>
</table>

ouch!
Bot hunting (Zeus)

ZeuS Tracker :: IP address 173.213.76.149

IP address: 173.213.76.149
Hostname: n/a
# of ZeuS Hosts: 8
# of active files: 16

SBL: 173.213.76.149
AS number: AS30693
AS name: EONIX-CORPORATION-AS-PHX01-WWW-INFINITIE-NET - Eonix Corporation
Country: United States (US)

Below is a list of all ZeuS Hosts which are currently hosted on this IP address.

Hosts on this IP address

<table>
<thead>
<tr>
<th>Dateadded</th>
<th>CC</th>
<th>RU</th>
<th>Host</th>
<th>Status</th>
<th>Files online</th>
<th>Registrar</th>
<th>Nameservers</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011-09-05</td>
<td>CC</td>
<td></td>
<td>utppvylfnkqsid.info</td>
<td>online</td>
<td>2</td>
<td>Directi Internet Solutions Pvt</td>
<td>dns1.epi, dns3.epi</td>
</tr>
<tr>
<td>2011-09-05</td>
<td>CC</td>
<td></td>
<td>jdwvlqsgqiwvxdkt.com</td>
<td>online</td>
<td>2</td>
<td>GODADDY.COM, INC.</td>
<td>ns35.dns35.com</td>
</tr>
<tr>
<td>2011-09-04</td>
<td>CC</td>
<td></td>
<td>vroxnsipqtmrlnq.blz</td>
<td>online</td>
<td>2</td>
<td>NAMESECURE,INC.</td>
<td>dns1.nai</td>
</tr>
<tr>
<td>2011-09-04</td>
<td>CC</td>
<td></td>
<td>jfjpdsqirhsyppqnn.org</td>
<td>online</td>
<td>2</td>
<td>NameSecure, L.L.C. (R58-LKOR)</td>
<td>dns1.nai</td>
</tr>
<tr>
<td>2011-09-03</td>
<td>CC</td>
<td></td>
<td>krifgkmckksqegol.biz</td>
<td>online</td>
<td>2</td>
<td>NAMESECURE,INC.</td>
<td>dns1.nai</td>
</tr>
<tr>
<td>2011-09-02</td>
<td>CC</td>
<td></td>
<td>aongrnervqnet.net</td>
<td>online</td>
<td>2</td>
<td>NAMESECURE.COM</td>
<td>dns1.nai</td>
</tr>
<tr>
<td>2011-09-02</td>
<td>CC</td>
<td></td>
<td>xqyikmnnshgxcpy.net</td>
<td>online</td>
<td>2</td>
<td>NAMESECURE.COM</td>
<td>dns1.nai</td>
</tr>
<tr>
<td>2011-09-02</td>
<td>CC</td>
<td></td>
<td>xernpyknfrfhklun.com</td>
<td>online</td>
<td>2</td>
<td>NAMESECURE.COM</td>
<td>dns1.nai</td>
</tr>
</tbody>
</table>

# of Host on this IP address: 8
Bot hunting (fast-flux)

ZeuS Tracker :: C&C indingo.ru

The list below shows all ZeuS configs, ZeuS binaries, ZeuS dropzones and other domains...

<table>
<thead>
<tr>
<th>Bailiwick</th>
<th>indingo.ru</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>93</td>
</tr>
<tr>
<td>first seen</td>
<td>2011-09-02 01:30:37 -0700</td>
</tr>
<tr>
<td>last seen</td>
<td>2011-09-04 03:47:38 -0700</td>
</tr>
</tbody>
</table>

| indingo.ru | A 60.19.30.134 |
| indingo.ru | A 60.19.30.135 |
| indingo.ru | A 61.197.232.43 |
| indingo.ru | A 63.226.215.202 |
| indingo.ru | A 78.196.237.195 |

<table>
<thead>
<tr>
<th>Bailiwick</th>
<th>indingo.ru</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>15</td>
</tr>
<tr>
<td>first seen</td>
<td>2011-09-02 12:26:03 -0700</td>
</tr>
<tr>
<td>last seen</td>
<td>2011-09-05 00:03:46 -0700</td>
</tr>
</tbody>
</table>

| indingo.ru | A 60.19.30.134 |
| indingo.ru | A 60.19.30.135 |
| indingo.ru | A 61.197.232.43 |
| indingo.ru | A 63.226.215.202 |
| indingo.ru | A 113.161.87.175 |

... more domains
... more IP resources
Spammers ❤ DNS

[162] [2011-09-06 05:31:35.########] [1:2 ISC email]
type: spamtrap
srchost: 117.yyy.yyy.yyy
bodyurl: hxxp://Despo.pharmacyramat.ru/?xxxxxxxxxxxxx
... redirects to “hxxp://www.medicostb.com/”
Data processing

- ISC Passive DNS Architecture (Edmonds)
  - https://kb.isc.org/article/AA-00654/
- Multiple relay upload servers robustly accept uploads and broadcast/replay them on SIE channels
- PassiveDNS processing server (48GB ram, CPU)
- DNSDB master server (12TB disk-based)
- DNSDB read replica (1.2TB SSD)
Law enforcement,
Security researchers,
CERTs, ISPs

Commercial and
public benefit efforts

Law enforcement,
Security researchers,
CERTs, ISPs
Making by-products available

Data reduction

Raw passive DNS – VLAN 202 – 100 Mbps.

First stage reduction – VLAN 207 – 5-10 Mbps.

Second stage reduction – VLAN 208 – 3-5 Mbps.

Third stage reduction – VLAN 204 – 1-2 Mbps.
Upload data (ISC:dnsqr)

[248] [2012-06-12 09:27:42.466236000] [1:9 ISC dnsqr] [NMSG_ID] []
type: UDP_QUERY_RESPONSE
query_ip: WW.XX.YY.ZZ
response_ip: 209.8.112.123
proto: UDP (17)
query_port: 22740
response_port: 53
id: 5875
qname: e319.g.akamaiedge.net.
qclass: IN (1)
qtype: A (1)
rcode: NOERROR (0)
delay: 0.000856
udp_checksum: CORRECT

query: [50 octets]
;; ->>HEADER<<- opcode: QUERY, rcode: NOERROR, id: 5875
;; flags: QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
e319.g.akamaiedge.net. IN A

response: [55 octets]
;; ->>HEADER<<- opcode: QUERY, rcode: NOERROR, id: 5875
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
e319.g.akamaiedge.net. IN A

 ;; ANSWER SECTION:
e319.g.akamaiedge.net. 20 IN A 184.24.193.107

 ;; AUTHORITY SECTION:

 ;; ADDITIONAL SECTION:

---

query: [50 octets]
;; ->>HEADER<<- opcode: QUERY, rcode: NOERROR, id: 5875
;; flags: QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
e319.g.akamaiedge.net. IN A

response: [55 octets]
;; ->>HEADER<<- opcode: QUERY, rcode: NOERROR, id: 5875
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

;; QUESTION SECTION:
e319.g.akamaiedge.net. IN A

 ;; ANSWER SECTION:
e319.g.akamaiedge.net. 20 IN A 184.24.193.107

 ;; AUTHORITY SECTION:

 ;; ADDITIONAL SECTION:

---
Tool chain (202->207->208)

nmsg-dns-cache
  --cache_mode front <--- deduplication of DNS RRSET responses
  --num_threads 8
  --cache_mem_size 16G
  --max_entry_duration 7200
  --max_input_age 3600
  --stats_frequency 60
  --spool [ch202]
  --write [ch207]
  --discard [ch206] <--- errors in input data

nmsg-dns-cache
  --cache_mode back <--- RRSET/bailiwick deduplication and verification
  --num_threads 8
  --cache_dir /srv/isc-passive-dns/cache
  --cache_mem_size 16G
  --max_entry_duration 21600
  --bwick_mem_size 16G
  --bootstrap_file /srv/isc-passive-dns/bootstrap/root.nmsg
  --stats_frequency 60
  --read [ch207]
  --write [ch208]
  --discard [ch206] <--- out-of-bailiwick data
Tool chain (208->204)

Three types of filtering: SOA, wildcards, regex

nmsg-dns-filter
   --discard_soa
   --dns_blacklist_file [dns_blacklist.txt]
   --regex_blacklist_file [regex_blacklist.txt]
   --read [ch208]
   --write [ch204]
   --filter [ch206]  <--- rrsets that failed soa or dns_blacklist_file

regex_blacklist example:    ^dhcp-[0-9]+\..*\.sql1\.isc\.org$

dns_blacklist example:      *.multi.surbl.org.
                        **.channel.facebook.com.
Data after processing (ch204)

[113] [2012-06-12 09:44:52.124765837] [2:1 SIE dnsdedupe] [NMSG-ID] [] []
type: INSERTION
count: 1
time_first: 2012-06-12 09:44:00
time_last: 2012-06-12 09:44:00
response_ip: 192.42.93.30
bailiwick: com.
rrclass: IN (1)
rrtype: NS (2)
rrttl: 172800

[103] [2012-06-12 09:41:18.051765837] [2:1 SIE dnsdedupe] [NMSD-ID] [] []
type: EXPIRATION
count: 18
time_first: 2012-06-12 01:41:37
time_last: 2012-06-12 06:58:20
bailiwick: com.
rrclass: IN (1)
rrtype: NS (2)
rrttl: 172800
rdata: ns1.savvis.net.
rdata: ns2.savvis.net.
rdata: ns3.savvis.net.
DNSDB (lessons earned)

- BerkeleyDB4 file (I/O bottleneck, data loss)
- MySQL (hash table, INSERT ON DUPLICATE, inserts got in way of queries, no god way to CIDR/Wildcard)
- PostgreSQL (liked CIDR range queries, but I/O ground to hal as index grew in size)
- Not scalable – too much I/O, uneven distribution
- MySQL + SSD + memcache – Could keep up with I/O, limited range functionality
- NoSQL – learned from MRTG rollups, sorting reverse domains to do CIDR and wildcard lookups quickly, time-range based HSM (memory, SSD, disk), good processing speed, lousy UI
DNSDB (evolve)

- 2010: Cassandra – clustered storage, removed single-server bottleneck, optimized for writes, web UI and http API interface – con: JRE, cashed from queries returning too many results

- 2011: TokyoCabinet – file-based storage, in-memory and SSD storage allowed reaction of read-optimized files that we could even export or scale with SSD-based server (price of SSD coming down, price of disk going up [floods])

- 2012: DnsTable – Robert created generic library/utility kit for sort-optimized key/value store (mtbl) then wrote utility wrappers for DNS-specific processing (dnstable) including web UI and http API access interface
  - Interesting: https://github.com/edmonds/mtbl
Some more background

- Robert Edmonds, “Passive DNS Hardening”
  - Video: http://bitly.com/lAJHVZ (DEFCON 18, Jul 2010)

- ISC Passive DNS and Privacy Whitepaper
  - Available upon request (dnsdb@isc.org) or soon at http://rsf.isc.org

- ISC Webinar, “SIE & Passive DNS”
  - Note: Shows examples of how PassiveDNS data has been provided to and used by several research efforts.
DNSDB API

```plaintext
$ DNSDB_FORMAT=json isc-dnsdb-query rdata ip 192.0.32.10 | sort
{"rrtype": "A", "rrname": "example.com.", "rdata": "192.0.32.10"}
{"rrtype": "A", "rrname": "example.edu.", "rdata": "192.0.32.10"}
{"rrtype": "A", "rrname": "example.net.", "rdata": "192.0.32.10"}
{"rrtype": "A", "rrname": "example.org.", "rdata": "192.0.32.10"}
{"rrtype": "A", "rrname": "mal1.gbs-clan.de.", "rdata": "192.0.32.10"}
{"rrtype": "A", "rrname": "mail2.gbs-clan.de.", "rdata": "192.0.32.10"}
{"rrtype": "A", "rrname": "scribble.co.uk.", "rdata": "192.0.32.10"}
{"rrtype": "A", "rrname": "www.example.com.", "rdata": "192.0.32.10"}
{"rrtype": "A", "rrname": "www.example.edu.", "rdata": "192.0.32.10"}
{"rrtype": "A", "rrname": "www.example.net.", "rdata": "192.0.32.10"}
{"rrtype": "A", "rrname": "www.example.org.", "rdata": "192.0.32.10"}
```

... for programmed lookups and cross-references and search.
... gets around web browser javascript limitations, too.

Restful API returns text or JSON with properly encoded URI representing query.
Documentation available here: https://dnsdb.isc.org/doc/isc-dnsdb-api.html
API CLI one-liner

$ dig medicostb.com ns
medicostb.com. 169386 IN NS ns2.dnsaq.ru.

$ ( for f in `isc_dnsdb_query.py -n ns1.upsdns.com.ua/NS | awk '{print $1}'`; do isc_dnsdb_query.py -r $f -j | egrep 'time_last": 1315[12]'; done ) | awk '{print $8}' | sort -u
"healthtr.com.",
"medicacpr.ru.",
"medicannk.com.",
"mediccker.ru.",
"mediccklr.ru.",
"medicehok.com.",
"medicelcr.ru.",
"medicellk.com.",
"medicemur.ru.",
"medicheek.com.",
"medichmar.ru.",
...etc...

Script isc_dnsdb_query.py is available at:
ftp://ftp.isc.org/isc/nmsg/misc
Who gets access?

- DNSDB User Interface or limited API key
  - Prefer vetted member of Operational Security community, but care more that you're at least not a bad guy.
  - Public benefit use
  - Most casual users query <1000 queries per day
  - Passive DNS contributors (submit data)
  - Expedited FIRST 24 registration:
    - See Eric during 3pm sessions this week. Bring ID and card.
  - After conference: https://dnsdb.isc.org/#Apply

- For higher query limits, commercial use
  - Get a limited key first, then contact <sales@isc.org> about upgrading.
  - Funds helps maintain the service and development. Anything extra is required to be spent by our parent 501(c)3 non-profit – more good work!
Even more

- Export of database on hourly/daily/monthly possible
- Real-time data feeds/by-products available
- We can teach you how to build your own
- We're considering open source model for programs that we use.
Community

- ISC:dnsqr can convert back to PCAP with a tool for incorporation into other projects. Why not benefit from hardening in our collection tools?

- CERTs or large ISPs worried about country privacy rules can build their own collectors and databases and share aggregated data with others (or ISC SIE). We've implemented two DNSDB systems outside of ISC.

- DNSDB is an example of one capability ISC has made available to the Internet security community. There's plenty more work and projects that we'd like to do. Consider supporting us as a Resiliency and Security Forum member: http://rsf.isc.org/
Questions?

• General DNSDB questions:
  • <dnsdb@isc.org>

• Applying:
  • https://dnsdb.isc.org/Apply

• Eric Ziegast <ziegast@isc.org>
  
  PGP: 7667 7BFB 3125 95EF B5B5 604A CD08 98D6 0BD0 D57D