Passive DNS Hardening

Robert Edmonds
Internet Systems Consortium, Inc.

Structure of this talk

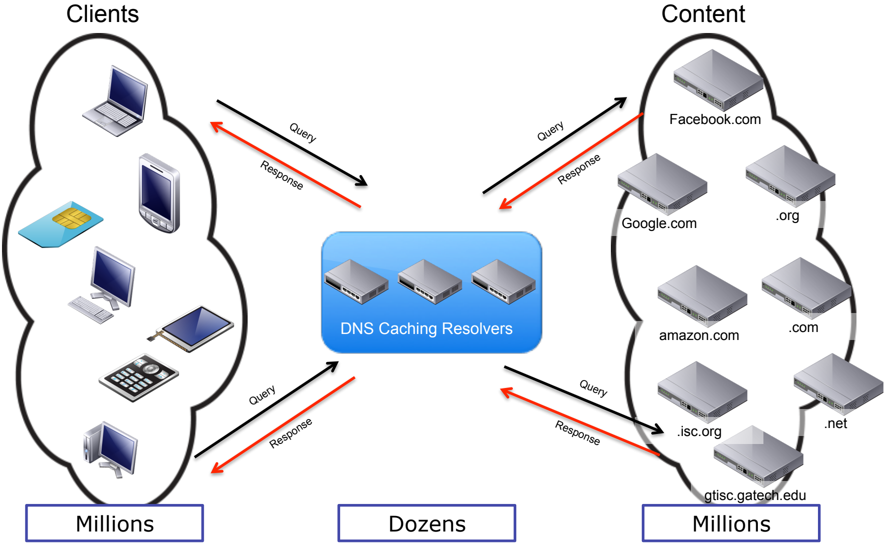
- Introduction
  - DNS
  - Passive DNS
  - ISC SIE
- DNS security issues
  - Kashpureff poisoning
  - Kaminsky poisoning
- Passive DNS security issues
  - Record injection
  - Response spoofing
- ISC DNSDB
  - Architecture
  - Demos

The Domain Name System

- "The DNS maps hostnames to IP addresses."
- More generally, it maps *(key, type)* tuples to a set of unordered *values*. again, we can think of the DNS as basically a multi-value distributed key-value store.

Clients, caches, content

- Clients request full resolution service from caches.
- Caches make zero or more inquiries to DNS content servers on behalf of clients. Results are cached for a limited time to serve future client requests.
- Content nameservers serve DNS records for zones that have been delegated to them.

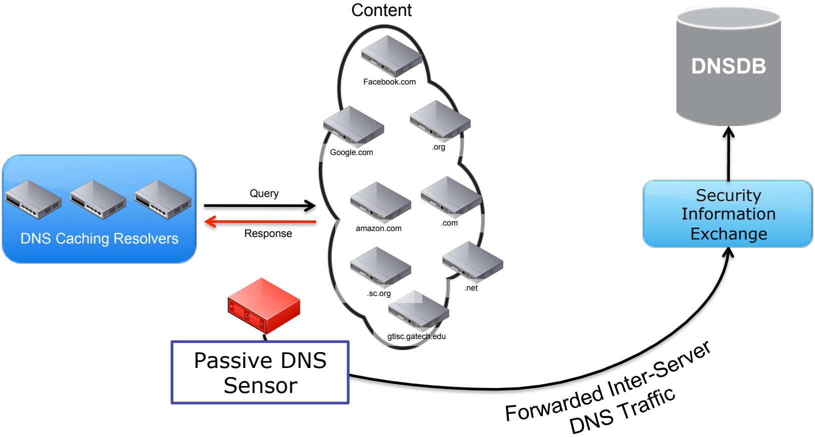

Clients
Content
Query
Response
Query
Response
Facebook.com
Google.com
.org
DNS Caching Resolvers
amazon.com
.com
Query
Response
Query
Response
.isc.org
.net
gtisc.gatech.edu
Millions
Dozens
Millions

Client-server and inter-server DNS protocols

- The DNS is actually two different protocols that share a common wire format.
  - The client-to-server protocol spoken between clients and caches.
  - The inter-server protocol spoken between caches and content servers.
- Passive DNS focuses on the latter.

Passive DNS

- Passive DNS replication is a technology invented in 2004 by Florian Weimer.
  - Many uses! Malware, e-crime, legitimate Internet services all use the DNS.
- Inter-server DNS messages are captured by sensors and forwarded to a collection point for analysis.
- After being processed, individual DNS records are stored in a database.


Content
Facebook.com
Google.com
.org
amazon.com
.com
.sc.org
.net
gtisc.gatech.edu
DNS Caching Resolvers
Query
Response
Passive DNS Sensor
DNSDB
Security Information Exchange
Forwarded Inter-Server DNS Traffic

Passive DNS deployments

- Florian Weimer's original `dnslogger`, first at RUS-CERT, then at BFK.de (2004–).
- Bojan Zdrnja's `dnsparse` (2006–).
- ISC's Security Information Exchange (2007–).

ISC Security Information Exchange

- SIE is a distribution network for different types of security data.
  - One of those types of data is passive DNS.
- Sensor operators upload batches of data to SIE.
- Data is broadcast onto private VLANs.
- `NMSG` format is used to encapsulate data.
  - Has a number of features which make it very useful for storing passive DNS data, but won't be covered further.
  - See our Google Tech Talk for more information: http://www.isc.org/community/presentations/video.

DNS Security Issues

- Passive DNS captures both signed and unsigned data, so DNSSEC cannot help us.
- What security issues are there in the DNS that are relevant to passive DNS?
  - Kashpureff poisoning
  - Kaminsky poisoning
    - (Actually, just response spoofing in general.)

Kashpureff poisoning

- Kashpureff poisoning is the name given to a particular type of DNS cache poisoning.
    - The attacker runs a content nameserver.
    - A client is enticed to lookup a domain name under the attacker's control.
    - The cache contacts the attacker's nameserver.
        - The attacker's nameserver provides extra records to the cache.
    - The extra records are inserted into the cache instead of being discarded.

Kashpureff poisoning example

```
Q: malicious.example.com.   IN A    ?

R: malicious.example.com.   IN NS   www.example.net.

R: www.example.net.         IN A    203.0.113.67
```

Kashpureff poisoning example

```
Q: malicious.example.com.   IN A    ?

R: malicious.example.com.   IN NS   www.example.net.

R: www.example.net.         IN A    203.0.113.67
```

Kashpureff poisoning example

```
Q: malicious.example.com.   IN A    ?

R: malicious.example.com.   IN NS   www.example.net.

R: www.example.net.         IN A    203.0.113.67
```

Kashpureff hardening

- 1997: Eugene Kashpureff hijacks the InterNIC website.
- `BIND 4.9.6` and `8.1.1` introduce hardening against Kashpureff poisoning.
- RFC 2181 is published.
    - See §5.4.1 “Ranking data” for details.

Lack of entropy

- 2000: DJB observes that a maximum of only about 31-32 bits of entropy can protect a UDP DNS query.
- Other DNS implementations slow to adopt SPR.
- 32 bits of entropy particularly weak for a session ID due to the **birthday attack** problem.
  - Newer protocols use cryptographically secure session IDs with 64, 128, or more bits.

Kaminsky poisoning

- 2008: Dan Kaminsky notices that the TTL can be bypassed.
- Coordinated, multi-vendor patches are released to implement source port randomization.
- SPR makes Kaminsky attacks harder, but not impossible.

Relevance to passive DNS

- Weimer's 2005 paper notes several problems with verifying passive DNS data.
- Kashpureff and Kaminsky poisoning of "active DNS" have analogues in passive DNS.
  - Passive DNS sensors can't see the DNS cache's "bailiwick", leading to **record injection**.
  - **Spoofed responses** are treated just like normal responses.
    - A **single** spoofed response can poison the passive DNS database!
- **Goal: make passive DNS *at least* as reliable as active DNS.**

Protecting the capture stage against response spoofing

- Capture both queries **and** responses.
- Correlate responses with previously seen queries.
- The DNS message **9-tuple**:
  1. Initiator IP address
  2. Initiator port
  3. Target IP address
  4. Target port
  5. Internet protocol
  6. DNS ID
  7. Query name
  8. Query type
  9. Query class

nmsg/dnsqr

- `dnsqr` is a message module for ISC's `libnmsg` specifically designed for passive DNS capture.
- UDP DNS transactions are classified into three categories:
  1. `UDP_QUERY_RESPONSE`
  2. `UDP_UNANSWERED_QUERY`
  3. `UDP_UNSOLICITED_RESPONSE`
- Performs IP reassembly, too!

Protecting the analysis stage against record injection

- Caches internally associate a “bailiwick” with each outgoing query.
- The cache knows what bailiwick to use, because it knows why it’s sending a particular query.
- We have to calculate the bailiwick ourselves.
- Protection against record injection requires protection against spoofed responses.
  - (Otherwise, an attacker could just spoof the record **and** the source IP address of an in-bailiwick nameserver.)

Passive DNS bailiwick algorithm

- Must operate completely **passively**.
- Must provide a boolean **true** or **false** for each record.
  - "For each record name, is the response IP address a nameserver for the zone that contains or can contain this name?"
- Example: `root` nameservers can assert knowledge about **any** name!
- Example: Verisign's `gtld` servers can assert knowledge about any domain name ending in `.com` or `.net`.

Passive DNS bailiwick algorithm

- Initialize bailiwick cache with a copy of the root zone.
    - Cache starts off with knowledge of which servers serve the root and TLDs.
- Find all **potential** zones that a name could be located in.
- Check whether any of the nameservers for those zones are the nameserver that sent the response.
- Each time an `NS`, `A`, or `AAAA` record is verified by the algorithm, it is inserted into the bailiwick cache.

Passive DNS bailiwick algorithm example

Name: `example.com.`
Server: `192.5.6.30`

- Potential zones:
  - `example.com.`
  - `com.`
  - `.`
- Zones in bailiwick cache:
  - `com.`
  - `.`
- Check: `example.com./NS`? Not found.
- Check: `com./NS`? Found 13 nameservers.
- Check: are any of them `192.5.6.30`? Yes.

Passive DNS bailiwick algorithm example

```
com.                 IN   NS   a.gtld-servers.net.
a.gtld-servers.net.  IN   A    192.5.6.30
```

Passive DNS bailiwick algorithm example

```
;; QUESTION SECTION:
;www.example.com.          IN   A

;; AUTHORITY SECTION:
example.com.          172800   IN   NS   a.iana-servers.net.
example.com.          172800   IN   NS   b.iana-servers.net.

;; ADDITIONAL SECTION:
a.iana-servers.net.   172800   IN   A    192.0.34.43
b.iana-servers.net.   172800   IN   A    193.0.0.236

;; SERVER: 192.5.6.30#53(192.5.6.30)
```

Passive DNS bailiwick algorithm example

```
;; QUESTION SECTION:
;www.example.com.                IN    A

;; ANSWER SECTION:
www.example.com.     172800      IN    A     192.0.32.10

;; AUTHORITY SECTION:
example.com.         172800      IN    NS    a.iana-servers.net.
example.com.         172800      IN    NS    b.iana-servers.net.

;; SERVER: 192.0.34.43#53(192.0.34.43)
```

Passive DNS bailiwick algorithm example

Name: `www.example.com.`

Server: `192.0.34.43`

- Potential zones:
  - `www.example.com.`
  - `example.com.`
  - `com.`
  - `.`
- Zones in bailiwick cache:
  - `example.com.`
  - `com.`
  - `.`
- Check: `www.example.com./NS`? Not found.
- Check: `example.com./NS`? Found 2 nameservers.
- Check: are any of them `192.0.34.43`? Yes.

DNSDB

- DNSDB is a database for storing DNS records.
    - Data is loaded from passive DNS and zone files.
    - Individual DNS records are stored in an Apache Cassandra database.
        - Offers key-value store distributed across multiple machines.
        - Good fit for DNS data.
        - Sustains extremely high write throughput because all writes are sequential.
    - Offers a RESTful HTTP API and web search interface.
- Database currently consumes about 500 GB out of 27 TB.

Architecture

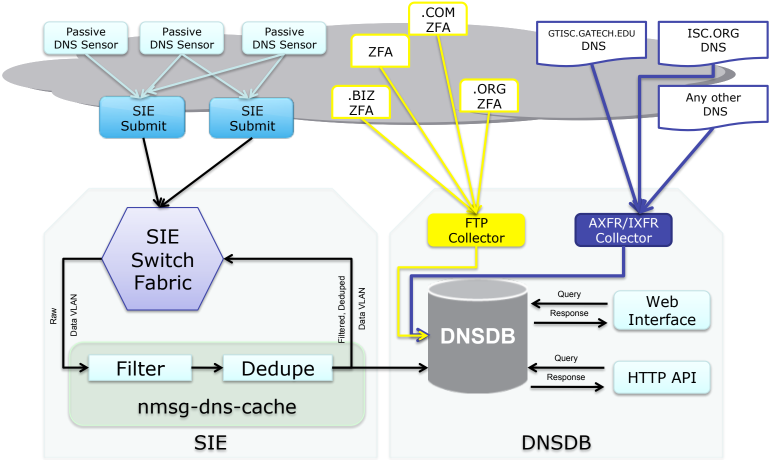
- Components
  - Data sources
    - `nmsg-dns-cache`
    - DNS TLD zones (FTP via ZFA programs): `com`, `net`, `org`, etc.
    - DNS zones (standard `AXFR`/`IXFR` protocol)
  - Data loaders
    - Deduplicated passive DNS
    - Zone file data

Data source: `nmsg-dns-cache`

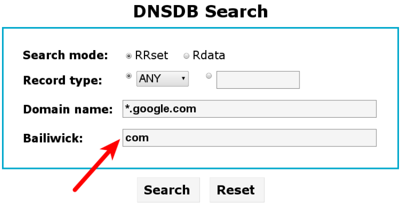
- Reads raw DNS responses from passive DNS.
- Parses each DNS message into individual DNS RRsets.
- Series of filters reduce the total amount of data by about 50%.
- RRsets are then inserted into an in-memory cache.
- Cache is expired in FIFO order.
- When RRsets expire from the cache, they form the final `nmsg-dns-cache` output.

Data source: zone files

- gTLD Zone File Access programs: `com`, `net`, `org`, `info`, `biz`, `name`
- AXFR'd zones: `isc.org`, a few other "test" zones.


Passive DNS Sensor
Passive DNS Sensor
Passive DNS Sensor
SIE Submit
SIE Submit
.COM ZFA
ZFA
.BIZ ZFA
.ORG ZFA
GTISC.GATECH.EDU DNS
ISC.ORG DNS
Any other DNS
SIE Switch Fabric
Raw
Data VLAN
Filtered, Deduped
Data VLAN
Filter
Dedupe
nmsg-dns-cache
SIE
FTP Collector
AXFR/IXFR Collector
DNSDB
Query
Response
Web Interface
Query
Response
HTTP API
DNSDB

Example #1: *.google.com



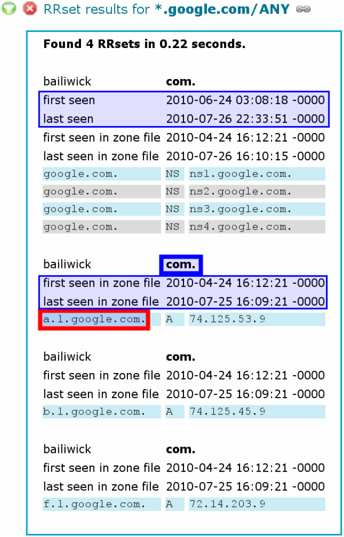
DNSDB Search

Search mode: ● RRset ○ Rdata

Record type: ● ANY ○

Domain name: *.google.com

Bailiwick: com



RRset results for *.google.com/ANY

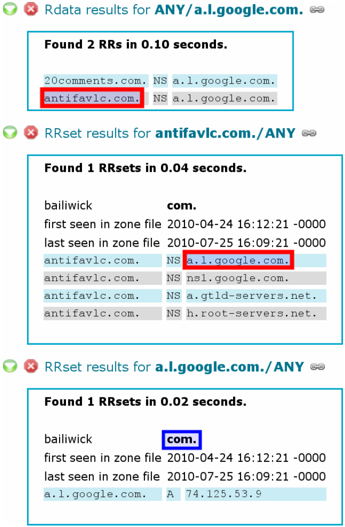
Found 4 RRsets in 0.22 seconds.

```
bailiwick                 com.
first seen                2010-06-24 03:08:18 -0000
last seen                 2010-07-26 22:33:51 -0000
first seen in zone file   2010-04-24 16:12:21 -0000
last seen in zone file    2010-07-26 16:10:15 -0000
google.com.               NS   ns1.google.com.
google.com.               NS   ns2.google.com.
google.com.               NS   ns3.google.com.
google.com.               NS   ns4.google.com.

bailiwick                 com.
first seen in zone file   2010-04-24 16:12:21 -0000
last seen in zone file    2010-07-25 16:09:21 -0000
a.l.google.com.           A    74.125.53.9

bailiwick                 com.
first seen in zone file   2010-04-24 16:12:21 -0000
last seen in zone file    2010-07-25 16:09:21 -0000
b.l.google.com.           A    74.125.45.9

bailiwick                 com.
first seen in zone file   2010-04-24 16:12:21 -0000
last seen in zone file    2010-07-25 16:09:21 -0000
f.l.google.com.           A    72.14.203.9
```



Rdata results for **ANY/a.l.google.com.**

Found 2 RRs in 0.10 seconds.

```
20comments.com.  NS  a.l.google.com.
antifavlc.com.   NS  a.l.google.com.
```

RRset results for **antifavlc.com./ANY**

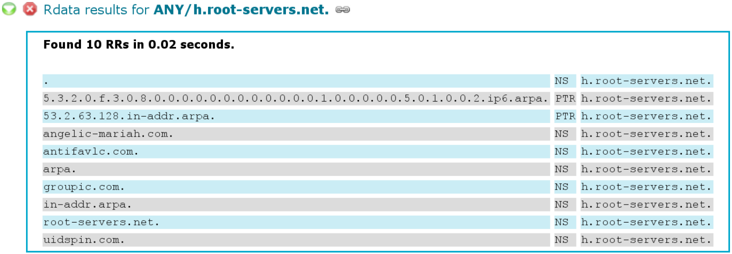
Found 1 RRsets in 0.04 seconds.

```
bailiwick            com.
first seen in zone file  2010-04-24 16:12:21 -0000
last seen in zone file   2010-07-25 16:09:21 -0000
antifavlc.com.       NS  a.l.google.com.
antifavlc.com.       NS  ns1.google.com.
antifavlc.com.       NS  a.gtld-servers.net.
antifavlc.com.       NS  h.root-servers.net.
```

RRset results for **a.l.google.com./ANY**

Found 1 RRsets in 0.02 seconds.

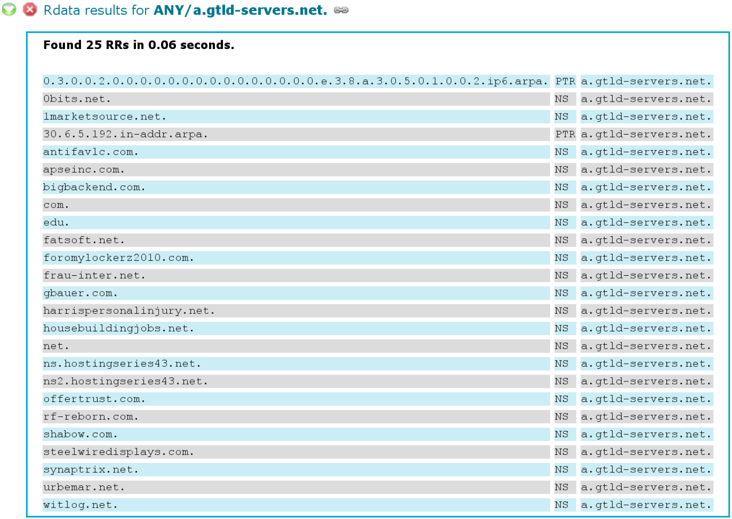
```
bailiwick            com.
first seen in zone file  2010-04-24 16:12:21 -0000
last seen in zone file   2010-07-25 16:09:21 -0000
a.l.google.com.      A   74.125.53.9
```



Rdata results for **ANY/h.root-servers.net.**

Found 10 RRs in 0.02 seconds.

.	NS	h.root-servers.net.
5.3.2.0.f.3.0.8.0.0.0.0.0.0.0.0.0.0.0.0.1.0.0.0.0.0.5.0.1.0.0.2.ip6.arpa.	PTR	h.root-servers.net.
53.2.63.128.in-addr.arpa.	PTR	h.root-servers.net.
angelic-mariah.com.	NS	h.root-servers.net.
antifavlc.com.	NS	h.root-servers.net.
arpa.	NS	h.root-servers.net.
groupic.com.	NS	h.root-servers.net.
in-addr.arpa.	NS	h.root-servers.net.
root-servers.net.	NS	h.root-servers.net.
uidspin.com.	NS	h.root-servers.net.



Rdata results for **ANY/a.gtld-servers.net.**

Found 25 RRs in 0.06 seconds.

0.3.0.0.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.e.3.8.a.3.0.5.0.1.0.0.2.ip6.arpa.	PTR	a.gtld-servers.net.
0bits.net.	NS	a.gtld-servers.net.
1marketsource.net.	NS	a.gtld-servers.net.
30.6.5.192.in-addr.arpa.	PTR	a.gtld-servers.net.
antifavlc.com.	NS	a.gtld-servers.net.
apseinc.com.	NS	a.gtld-servers.net.
bigbackend.com.	NS	a.gtld-servers.net.
com.	NS	a.gtld-servers.net.
edu.	NS	a.gtld-servers.net.
fatsoft.net.	NS	a.gtld-servers.net.
foromylockerz2010.com.	NS	a.gtld-servers.net.
frau-inter.net.	NS	a.gtld-servers.net.
gbauer.com.	NS	a.gtld-servers.net.
harrispersonalinjury.net.	NS	a.gtld-servers.net.
housebuildingjobs.net.	NS	a.gtld-servers.net.
net.	NS	a.gtld-servers.net.
ns.hostingseries43.net.	NS	a.gtld-servers.net.
ns2.hostingseries43.net.	NS	a.gtld-servers.net.
offertrust.com.	NS	a.gtld-servers.net.
rf-reborn.com.	NS	a.gtld-servers.net.
shabow.com.	NS	a.gtld-servers.net.
steelwiredisplays.com.	NS	a.gtld-servers.net.
synaptrix.net.	NS	a.gtld-servers.net.
urbemar.net.	NS	a.gtld-servers.net.
witlog.net.	NS	a.gtld-servers.net.