DNS Inconsistency

John Kristoff <jtk@depaul.edu>
Balajee Vamanan <bvamanan@uic.edu>
Chris Kanich <ckanich@uic.edu>
Introduction

• Parent NS RRset \( p \)
  example 1D IN NS ns1.example.edu.
  example 1D IN NS ns2.example.edu.

• Child NS RCset \( c \)
  example 1D IN NS ns1.example.edu.
  example 1D IN NS ns2.example.edu.
  example 1D IN NS ns3.example.edu.
Real, Badly Inconsistent Example

- Child NS RRset *cdm.depaul.edu*

```
cdm.depaul.edu.  3600 IN NS  ns1.cti.depaul.edu.
cdm.depaul.edu.  0     IN NS  shemp.cti.depaul.edu.
cdm.depaul.edu.  3600 IN NS  ns-colo.cti.depaul.edu.
cdm.depaul.edu.  3600 IN NS  dc-colo-cti.cti.depaul.edu.
cdm.depaul.edu.  3600 IN NS  bach.cti.depaul.edu.
cdm.depaul.edu.  3600 IN NS  ellington.cti.depaul.edu.
cdm.depaul.edu.  3600 IN NS  moe.cti.depaul.edu.
cdm.depaul.edu.  3600 IN NS  mozart.cti.depaul.edu.
```

```
  ...
ns-colo.cti.depaul.edu.   AAAA 2002:d8dc:b452::dbdc:b452
dc-colo-cti.cti.depaul.edu.  A    10.128.30.2
```
Conjecture

DNS infrastructure (NS parent/child RRset) inconsistency arises from asynchronous and uncoordinated NS RRset configuration
Methodology: .edu traversal

• Obtain all .edu names using `whois *` hack

```python
for each name in edu
    mark root_servers as visited
    get NS_RRset from an .edu NS for the name
    for each S in NS_RRset
        do_query( name, S )

do_query:
    return if already queried S for name
    mark ( name, S ) as visited
    get new_NS_RRset for name from S
    for each S* in new_NS_RRset
        do_query( name, S* )
```
## NS name mapping ambiguity

<table>
<thead>
<tr>
<th>name</th>
<th>IPv4 address</th>
<th>IPv4 address set</th>
<th>IPv6 address</th>
<th>IPv6 address set</th>
<th>IPv4 address + IPv6 address</th>
<th>IPv4 address set + IPv6 address</th>
<th>IPv4 address + IPv6 address set</th>
<th>IPv4 address set + IPv6 address set</th>
</tr>
</thead>
</table>
# Evaluating Inconsistency

<table>
<thead>
<tr>
<th>Error Type</th>
<th>Description</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>bad type (e.g. CNAME)</td>
<td></td>
</tr>
<tr>
<td>error</td>
<td>bad rdata (e.g. Ipaddr for NS)</td>
<td>29 (0.01)</td>
</tr>
<tr>
<td>error</td>
<td>TTL disagreement in NS RRset</td>
<td>141 (0.06)</td>
</tr>
<tr>
<td>error</td>
<td>DNSSEC validation failure</td>
<td></td>
</tr>
<tr>
<td>error</td>
<td>timeout/unreachable transient (e.g. down time)</td>
<td></td>
</tr>
<tr>
<td>error</td>
<td>timeout/unreachable permanent (e.g. misconfiguration)</td>
<td>1403 (6)</td>
</tr>
<tr>
<td>query_response</td>
<td>NOERROR</td>
<td>21593 (90)</td>
</tr>
<tr>
<td>query_response</td>
<td>NXDomain</td>
<td>23 (0.01)</td>
</tr>
<tr>
<td>query_response</td>
<td>REFUSED</td>
<td>679 (3)</td>
</tr>
<tr>
<td>query_response</td>
<td>SERVFAIL / FORMERR / NOTIMP / ...</td>
<td>142 (0.06)</td>
</tr>
<tr>
<td>query_response</td>
<td>referral after a referral</td>
<td>77 (0.03)</td>
</tr>
<tr>
<td>query_response</td>
<td>aa==0 when aa==1 expected</td>
<td>977 (4)</td>
</tr>
<tr>
<td>query_response</td>
<td>malicious or incorrect data</td>
<td></td>
</tr>
</tbody>
</table>
Parent/Child NS RRset Consistency

- parent=child
- parent+
- child+
- mismatch
Namespace != Infrastructure Graph
# Resolver (in)Stability

<table>
<thead>
<tr>
<th>Resolver</th>
<th>Distribution</th>
<th>Avoidance</th>
<th>Recovery</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIND</td>
<td>proportional</td>
<td>no</td>
<td>&lt; 1 sec</td>
</tr>
<tr>
<td>PowerDNS</td>
<td>spike dist.</td>
<td>no</td>
<td>3 min</td>
</tr>
<tr>
<td>Unbound</td>
<td>uniform</td>
<td>yes</td>
<td>15 min</td>
</tr>
<tr>
<td>DNSCache</td>
<td>uniform</td>
<td>no</td>
<td>&lt; 1 sec</td>
</tr>
<tr>
<td>WindowsDNS</td>
<td>uniform</td>
<td>yes</td>
<td>1 sec</td>
</tr>
</tbody>
</table>

- Source: Yu et al., Authority server selection in DNS cache resolvers, ACM SIGCOMM CCR 2012
- NOTE: negative caching => bursts of repeated failures
Discussion

• Inconsistency increases down the name space
• Inconsistency could exacerbate security threats
• Inconsistency may affect performance
• Inconsistency may lead to non-determinism
Questions

- Are some NS infrastructure graphs unknowable?
- Should consistency be encouraged? If so, how?
- There is no up/down sync, should there be?
- Should minimal-responses be preferred?
- Should repeated failures influence retry algorithms?
- Should NS RRs have had IPaddrs as RDATA?
- Is inconsistency worth studying further?