DNS: Defense and Attack

Paul Ebersman
pebersman@infoblox.com, @paul_ipv6
RIPE66, Dublin, 13-17 May 2013
DNS is you
DNS is who you are on the internet

- If your DNS zone isn’t available:
  - No email
  - No website
  - No internet services…
Robust DNS is worth money

Even managers/executives now see value of robust DNS
- Registry/Registrar security
- Who owns nameservers
- Who can update zone data and how
Attacking your cache
What is it?
- Inducing a name server to cache bogus records

Made possible by
- Flaws in name server implementations
- Short DNS message IDs (only 16 bits, or 0-65535)

Made easier on
- Open recursive name servers

Consequence
- Man in the middle attacks
Amit Klein of Trusteer found that flaws in most versions of BIND’s message ID generator (PRNG) don’t use sufficiently random message IDs

- If the current message ID is even, the next one is one of only 10 possible values
- Also possible, with 13-15 queries, to reproduce the state of the PRNG entirely, and guess all successive message IDs
Birthday Attacks

- Barring a man in the middle or a vulnerability, a hacker must guess the message ID in use
  - Isn’t that hard?
  - As it turns out, not that hard

- **Brute-force guessing is a birthday attack:**
  - 365 (or 366) possible birthdays, 65536 possible message IDs
  - Chances of two people chosen at random having different birthdays:

\[
\frac{364}{365} \approx 99.7\%
\]

- Chances of n people \((n > 1)\) chosen at random all having different birthdays:

\[
\bar{p}(n) = \frac{364}{365} \times \frac{363}{365} \times \ldots \times \frac{366 - n}{365} \quad p(n) = \left(1 - \bar{p}(n)\right)
\]
## Birthday Attacks (continued)

<table>
<thead>
<tr>
<th>People</th>
<th>Chances of two or more people having the same birthday</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>12%</td>
</tr>
<tr>
<td>20</td>
<td>41%</td>
</tr>
<tr>
<td>23</td>
<td>50.7%</td>
</tr>
<tr>
<td>30</td>
<td>70%</td>
</tr>
<tr>
<td>50</td>
<td>97%</td>
</tr>
<tr>
<td>100</td>
<td>99.999996%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Number of reply messages</th>
<th>Chances of guessing the right message ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>~20%</td>
</tr>
<tr>
<td>300</td>
<td>~40%</td>
</tr>
<tr>
<td>500</td>
<td>~80%</td>
</tr>
<tr>
<td>600</td>
<td>~90%</td>
</tr>
</tbody>
</table>
The Kashpureff Attack

- Eugene Kashpureff’s cache poisoning attack used a flaw in BIND’s additional data processing
The Kaminsky Vulnerability

- How do you get that many guesses at the right message ID?
The Kaminsky Vulnerability (continued)

- How does a response about q00001.paypal.com poison www.paypal.com’s A record?
- Response:

```
;; -->>HEADER<<- opcode: QUERY, status: NOERROR, id: 61718
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1,
ADDITIONAL: 1

;; QUESTION SECTION:
;q00001.paypal.com. IN A

;; AUTHORITY SECTION

;; ADDITIONAL SECTION
www.paypal.com. 86400 IN A 10.0.0.1
```
Saved by the Second Law of Thermodynamics

- To make it more difficult for a hacker to spoof a response, we use a random query port
  - In addition to a random message ID
  - If we use 8K or 16K source ports, we increase entropy by 13 or 14 bits
  - This increases the average time it would take to spoof a response substantially

- However, this is not a complete solution
  - Spoofing is harder, but still possible
  - Evgeniy Polyakov demonstrated that he could successfully spoof a patched BIND name server over high-speed LAN in about 10 hours
Defending your cache
Defenses

- More randomness in DNS msg IDs, source ports, etc.
- Better checks on glue
- DNSSEC
Overwhelming your authoritative servers
Sheer volume

- Botnet attacks in 10s of Gb’s
High Yield Results

- Asking for DNSSEC records
- Using NSEC3 against you
How to defend your servers
Harden your server

- Perimeter ACLs
- Higher capacity servers
- Clusters or load balanced servers
Spread yourself out

- Fatter internet pipes
- More authoritative servers (up to a point)
- Anycast
- HA
DNS use by the bad guys
DNS use by bad guys

- Command and control
- DNS Amplification
- Fastflux
  - single flux
  - double flux
- Storm, Conficker, etc.
Protecting your users via DNS
Dealing with malware

- Prevent infections (antivirus)
- Block at the perimeter (NGFW, IDS)
- Block at the client (DNS)
RPZ DNS

- Uses a reputation feed(s) (ala spam)
- Can be IP or DNS based ID
- Fast updates via AXFR/IXFR
- Protects infected clients, helps ID them
- Can isolate infected clients to walled garden