Ten DNS things you probably did not know about

How to bore your friends and amaze your geeks

Bert Hubert

POWERDNS

PowerDNS

- PowerDNS.COM BV gestart in 1999
- Eerste database gestuurde nameserver
  - Meeste oplossingen werken nog steeds van textfiles
- In gebruik van kleine tot zeer grote installaties
  - 10 miljoen+ domein installatie op twee servers (authoritative)
  - 20 miljoen gebruikers access provider (resolver)
- Open Source sinds 2002
- Gevestigde oplossing
  - >50% van .DE domeinen, niet onaardige hoeveelheid in Nederland
- “Community” & Commercieel gestuurd

1: DNS is binary clean, and has no problems with $-_#; in records!

- “everybody” knows you can't have a _ in a host name
- Especially Windows servers like to have an underscore however!
- However.. DNS has no problem with this.
- DNS is “binary clean”
  - AND case insensitive
- So go blame someone else ;_)
- (Paul Vixie)

Blame sendmail!

I outlawed _ as a side effect of “punting”. in order to strip/prevent newline characters in PTR targets, i had to be able to refer to an RFC (lest people come to me with many individual sob stories about this or that special character that either should or should not be stripped/prevented in gethostbyaddr().) the only RFC i found that had any remote chance of getting me off this hook was #952. ergo, _ had to die in order that my inbox might live.

but it was wrong, and the need for it is past, and it's time for redress

- paul vixie
http://www.mail-archive.com/nanog@merit.edu/msg32795.html

2: ANY queries are not ANY

- DNS Records can have many types
  - A
  - AAAA
  - MX
  - PTR
  - CNAME (yuck)
- For some reason, support was added to query all types at once, the 'ANY' query – type 0xff
- Authoritative servers do indeed support this
- Resolvers however don't! Because of weird wording in RFC 1034, they only return what is in the cache at that time!
- This means you can't use ANY queries for anything serious!
  - Not even to get A and AAAA addresses in one go..
- You can use ANY queries to crash nameservers though!

3: Well implemented DNS is "Secure"

- Much has been made of how easy it is to spoof DNS
  - Kaminsky etc
- And if you could, the results would be TREMENDOUS
- Think about redirecting everything ending on .com to your own servers..
  - My!
  - (xs4all could use their f.root to achieve the same ;-))
- Turns out that attractive as this is, and it requires only 50megabits/s to launch a credible attack, *it does not happen* with any frequency
  - Slow attack however..

4: DNS is secure.. if you do everything right

- In practice DNS is actually pretty secure
- But make one mistake, and it isn't..
- Little mistakes like:
  - Not answering invalid questions (!)
  - Sending multiple equivalent questions simultaneously
    - Forgetting that DNS is case insensitive, so sending equivalent questions without meaning to..
  - Using an insufficient amount of source ports to send queries from
    - Possibly because you are behind NAT
  - Enabling Path MTU discovery (!)

5: Fun with embedded NULLs and .

- (this is pretty new!)
- DNS uses rather clever 'binary clean' encoding:
  - www.xs4all.nl → \x3www\x6xs4all\x2nl**x0**
- This means that 'www.xs4all.nl**\x0**evilhost.nl' is a valid DNS name within the 'evilhost.nl' zone
- **On the wire** however, this will appear like a query for www.xs4all.nl (tcpdump, wireshark etc stop at the first null!)
- The same goes for “10www.xs4all2nl” which in tcpdump looks like a query for 'www.xs4all.nl', but to the nl servers looks like a query for 'www\.xs4all'.

5: fun with embedded NULLs and .

- So, why is this fun?
- Especially the embedded NULL allows you to hide your true queries from almost all monitoring tools (perhaps as to disguise your DNS tunnel!)
- More fun: as said before, if you can force a resolver to have multiple questions outstanding that it treats equivalently, you can spoof away
- I wonder if all nameservers see the difference between www.xs4all.nl**x0**.1.targetdomain.com and www.xs4all.nl**x0**.2.targetdomain.com
  - (no) ;-)
- Probably lots more you can do with this..

6: DNSSEC does not protect .SE, .BR, .ORG, .GOV etc!

- We could spend a day talking about the merits or lack thereof of DNSSEC
  - I am on a registry council these days tasked with seeing how DNSSEC can be implemented however, so at least I'm involved ;-)
- DNSSEC is not like 'https', 'DNSs'
  - Although the use of SSL is debatable too..
- DNSSEC protects the final, authoritative DNS message, so it is content protection and not transport protection
- This means that the delegation from xs4all.org to the xs4all nameservers (which is not authoritative!)
- If xs4all.org itself is not signed, there has been **zero gain**.
- DNSSEC only 'clicks' if everyone takes part
  - Ok, DNSSEC **does** protect the .org SOA record! Yay!

7: >95% of all queries arriving at the root server are bogus

- If there were no caches in DNS resolvers, each query would start at the root servers
- Luckily, we have caches, so they only need to tell you where the .com servers are once every day or so
  - .. in theory
- In practice, the world sucks, and over 95% (98% in one survey) of all queries arriving at the root servers are junk
- For example, .local, .lan, .office, “1.2.3.4”
- Printers asking a thousand queries/s..

8: There are over a hundred root-servers!

- As the Wikipedia, the fount of knowledge, still claims: “There are thirteen root server clusters that are authoritative for queries to the global DNS root zone”
- A-M.ROOT-SERVERS.NET
- This was a rather scary situation
  - I've visited the fake 'A.ROOT-SERVERS.NET' @ Verisign (real one was next to it)
  - USGOV was powerful back then..
- Slowly over the years, 'ANYCAST' was introduced, so many many servers now share the same IP addresses
- dig +norec @F.ROOT-SERVERS.NET HOSTNAME.BIND CHAOS TXT

9: Fun with fragments (sit tight!)

- (rather new)
- DNS responses are matched to their query based on source/destination port and query ID, together 32-bits of “security”
  - Turns out to be enough, but 16 bits is not
- Packets on the internet can be 1500 bytes
- DNSSEC requires longer 'datagrams', spread out over multiple packet fragments
- Secondary fragments do not contain the query ID, but do contain the 'IP ID', plus the checksum of the entire fragment has to match (also 16 bits)
- So, spoofing in the second fragment also requires 32 bits.. or does it?
- If you could spoof, you could have lots of fun!

9: fun with fragments (sit tight!)

- So.. to spoof in the second fragment, it should match up with the first
- Port numbers and id are in the first fragment, no need to match those!
  - Yay!
- Should match IPID, and the final checksum should match
  - Bummer, 32 bits!
- HOWEVER! We know the '16 bit one's complement' sum of the REAL second fragment
- And since the “internet checksum” is commutative, the final checksum will then also match!
- Race is now on to see how this can be used for fun and profit
  - Will surely harm DNSSEC reliability

10: All nameservers are made in .NL

- DJBDNS/TINYDNS: Dan Bernstein, lives in Eindhoven, The Netherlands
- BIND10: Project lead Shane Kerr, lives in Amsterdam, The Netherlands
- NSD: NLNETLabs, Amsterdam, The Netherlands
- Unbound: NLNETLabs, Amsterdam, The Netherlands
- PowerDNS: Rijswijk, The Netherlands
- **Nominum: Redwood City, CA, USA**

If it ain't Dutch, it ain't much! ;-)