DNSSEC - Maybe it's the Journey, not the Destination

Edward Lewis, NeuStar Inc.

APTLD

Monday, February 25, 2008
My perspective

- 1994   DNSSEC starts (before my time)
- 1996   Join in, writing code, definitions
- 1998   First "deploy how?" meeting
          DARPA, ISC, TIS @ IETF in LA
- 1999-'03 Ran technical workshops
          '99 in SE, 1st public hands-on
- 2002-'04 Work in an RIR
- 2004-'08+ Work in a Domain Name Registry
What I have seen

- Sound technology, adjusted as original requirements solidified
- As time went by
  - Requirements met by cheaper alternatives
  - Advancements from DNSSEC spin-offs
- Nobody (well, not many) wants to use it
Questions

If the problem is real,
If the technology solves the problem,
If people are promoting the technology

Why isn't it in use?
Was it a waste of time?
Hypothesis

• DNSSEC isn't green lighted because
  – It’s not much of a gain
    • Cache poisoning is no longer "the threat"
    • Software has gotten better
    • Specifications have gotten better
  – Modes of attack have moved on
• But DNSSEC “was” a success
Diminishing Returns
If there was a way to measure security, we'd see this

It's not that DNSSEC got worse.

Security Gap of DNS

Security Level of DNS

1994 0% Secure 2008

100% Secure

DNS has gotten better

APTLD, Feb 25, 2008
ed.lewis@neustar.biz
What was (once) wrong?

• Cache Poisoning
  – The main threat
• Buggy Software
  – Less known but more problematic
• Buggy Specifications
  – Documents were unclear
Cache Poisoning

• Spreading “wrong” answers was easy
  – Name servers were gullible
  – Asking "what the address of my school" could be answered with "oh, and here's the address of the tax office too."
  – Name server accepted all info seen, regardless of source
What beat cache poisoning?

• An RFC published in 1996 introduced a scale of "trustworthiness."
  – Disregard answers not pertinent to the question
• Current work on message ID forging
• Not foolhardy, but greatly reduced impact of cache poisoning attempts
  – Without the need of cryptography
Buggy software

• Buggy software made any security analysis hard, “is ‘it’ a:”
  – Protocol weakness
  – Code weakness
  – Protocol hack to avoid a code bug

• Code needed “professional help”
New Software

• DNSSEC directly caused a complete rewrite of BIND
  – Funding to get DNSSEC going covered BIND 9
• And many other factors contributed to new implementations
• Better code, fewer vulnerabilities
Buggy Specifications

• Early documents were engineering descriptions and not specifications
  – Language was not specific
  – Too much was assumed
  – Compliance never quantified

• Still today “original intents” are hotly debated
Clarifications

- The IETF has produced and is working on more clarifications
- Some fine details of DNS had never been worked on before DNSSEC
- Cleaner specifications lead to better implementations, and again fewer vulnerabilities
DNSSEC at a Crossroads

• After so long in development
  – Old problems have gone away
  – New ones appear
  – There's a history of gains
  – Different solutions are needed now
Today’s Threats

• Recent DNS “attacks” usually are explained by
  – Use of antiquated software
  – Denial of Service (DDoS or DoS)
  – Registration problems
  – Hiding true destinations of URLs from user
  – Malware on host changing DNS settings
Risk

• There's a risk that an authorized action will be incorrectly denied
  – A "name" seems to be "down"
  – One bad key could take down many names
• Poorly written security software may be the problem, beyond control
• Introduction (development) has risks
When is risk acceptable?

- Risk is acceptable if the action
  - is vital to some other core need
  - will sufficiently lower operating costs
  - promises a sufficient return on the investment
DNSSEC and Risk

- Doesn't solve the problems of today
- Operation of DNS made more costly
- There's no revenue-promising demand

- A "well do it anyway" attitude isn't prudent, too much is riding on the Internet now
Where Should Energy Go?

• Making sure software is updated
• Registration practices
• Traffic (basic UDP) management
• Application safety
• Host security
Summary

• DNSSEC may have already given us its benefit
  – New software, operations, specifications
• Does it still make sense to deploy DNSSEC?
• Should we focus on other issues?