Validating Caching Resolver

Wouter Wijngaards
proxied by
Jaap Akkerhuis
Introduction

• Why a new resolver?
  – Code diversity in DNS server monoculture
  – Alternative validator choice for BIND 9

• Deployment targets
  – Workgroup local DNS resolvers
  – Large caching resolver installations (ISP)
  – Validating library for applications

• About NLnet Labs
  – A not for profit, public benefit foundation
  – Developed NSD; DNSSEC aware, high performance authoritative name server
  – LDNS; Library to simplify DNS programming
Java Prototype

- DNS Server Features
  - IPv4 and IPv6, Recursion, Caching, Thread support, Access control for DNS service: not open recursor
  - DNSSEC validation, NSEC, NSEC3, ready for SHA256
  - Trust anchors: feature rich
  - Authoritative server: Absent
- Full Documentation and Some Tools
  - Unbound-checkconf
  - Unbound-host: validated host lookup
Features: Paranoia

• Forgery resilience: *full featured*
  – Scrubber filters packets for out-of-zone content
  – Follows RFC2181 trust model
  – Follows all recommendations from dnsop draft
    • Query name matching
    • Strong random numbers for ID
    • UDP source port random
    • IP source address random
    • RTT banding
Performance

Scenario: recursion (no query twice)
Server OS: Ubuntu 6.10
CPU: AMD Athlon 2400+, Mem: 1.5 GB
Network: RTL8169S 1000BaseTX
Q&A time

???

• Unbound – Validating Caching Resolver
  – Open source: BSD license
  – Portable: Linux, *BSD, Solaris, MacOS/X
  – Windows Port (POC)
    • DNSSEC for the masses

• Get 1.0.0 at http://unbound.net