Infrastructure for Experimental Replay and Mutation of DNS Queries

Liang Zhu, *John Heidemann*

joint work with Wes Hardaker, Terry Benzel

University of Southern California / Information Sciences Institute

at CAIDA / AIMS Workshop / San Diego, 2016-03-02

Copyright © 2017 by John Heidemann
Release terms: CC-BY-NC 4.0 international

USC Viterbi School of Engineering Information Sciences Institute

Challenge

- Given a new idea about DNS…
  - privacy: TLS or DNScrypt or something else?
  - does qname minimisation need optimizations?
  - location: Client Subnet or EIL or something else?
  - can we improve response to stresses like DDoS?
- how do we test it?
  - under *real conditions* today?
  - under *potential conditions* tomorrow?
- rigorously
  - believed by peer-reviewers
  - and operators
  - and policy makers

Our Approach: Trace Replay

- to explore "what if" scenarios with real data
- modeling is great, but often not definitive
  - DNS caching is really hard to model
  - and implementions vary from ideal

accurate, high-speed trace replay
is essential to study many open questions

Challenge: Scale (in *Many* Dimensions)

- many zones
- multiple levels of DNS hierarchy
- high rate queries
- diverse query sources
  - different RTTs (RTT matters!)

Design Requirements

- Avoiding traffic to the Internet
- Emulate complete DNS hierarchy, efficiently
- Manipulate queries arbitrarily
- Support multiple protocols
- Support high query rates accurately

Avoiding Traffic to the Internet

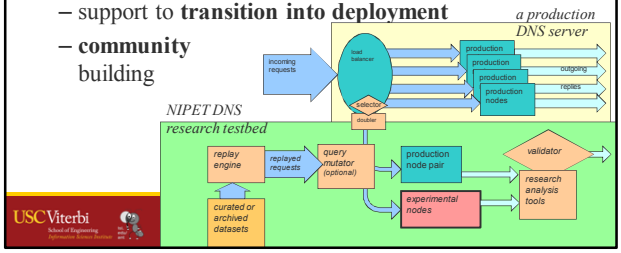
- challenge
  - reproducability (same every time)
  - experiment shouldn't stress real world
    - replays can be *large* and *repeated*
- our approach
  - convert traces to zones
  - fill in missing data (absent due to caching)
  - host synthetic zones locally
  - (challenge: variant responses from servers)

Emulating the Hierarchy, Efficiently: the problem

- challenge:
  - the DNS hierarchy matters
    - "." -> com. -> example.com. -> mail.example.com.
    - may see 100 to 1000s of zones in a short trace
  - efficiency matters
    - cannot us 100s to 1000s of computers (or even VMs or containers)
- observation
  - one DNS server *can* host many zones
  - problem: one server takes shortcuts
    - one server hosting host . and com. and example.com
    - if you ask for mail.example.com, it answers right away, skipping the round-trips to root and com and example.com

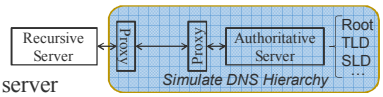
Naming and Internet Protocol Experimental Testbed (context)

- we're hoping to build a DNS research testbed
  - **experiments** on live data
  - access to **historical data**
  - support to **transition into deployment**
  - **community** building



Emulating the Hierarchy, Efficiently: the solution

- insight:
  - *split-horizon* DNS lets one server host many zones
  - *proxies* chose the horizon
- result:
  - efficient: one server with many zones
  - *and* correct: emulation gets right round-trips



Conclusions

- open-source software at https://ant.isi.edu/software/ldplayer
  - replay component already available
  - expect to release zone creator shortly
- datasets: https://ant.isi.edu/datasets/ and in https://impactcybertrust.org/
- towards a testbed: see https://ant.isi.edu/nipet/
- would love feedback
  - both about DNS trace replay
  - and broader idea of Research Infrastructure for DNS

Example Use Case: DNSSEC Key Sizes

- q: what if we change DNSSEC key sizes?
  - more traffic—we confirm *how* much more
  - confirm prior hard-coded sim (Wessels) with general mechanism
- q: what if *everything* was DNSSEC?
  - much more traffic (we quantify)
  - done with *query manipulation*