



Unbound

Validating Caching Resolver

Wouter Wijngaards

wouter@nlnetlabs.nl (NLnet Labs)

Overview

- Introduction: *Why another resolver?*
- Features
 - Anchors and Authority
 - Paranoia
- Design
- Tests
 - Cache performance
 - Recursion performance
- Summary

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

Development History

- The first architecture and a Java prototype was developed between 2006-2007.
 - Matt Larson,
 - David Blacka
 - Bill Manning
 - Geoff Sisson,
 - Roy Arends
 - Jacob Schlyter
- Current release candidate 0.11
 - Release of *1.0* expected within a month
 - Substantive testing and feedback of this and earlier versions by:
 - Alexander Gall (switch.ch)
 - Ondřej Surý (.cz)
 - Kai Storbeck (xs4all.nl)
 - Randy Bush (psg, iij)
- NLnet Labs joined early 2007
 - porting the prototype to C and taking on maintenance.
 - First public development release on <http://unbound.net/> in jan 2008



EP.NET

nominet

kirei

Features: Basic

- DNS Server
 - Recursion
 - IPv4 and IPv6 dual stack support
 - Access control for DNS service: not open recursor
 - DNSSEC validation
 - NSEC, NSEC3, ready for SHA256
- Tools
 - Unbound-checkconf
 - Unbound-host: validated host lookup
- Documentation
 - man pages, website and in code (doxygen)
- Thread support (optional): scalable performance

Features:

Anchors and Authority

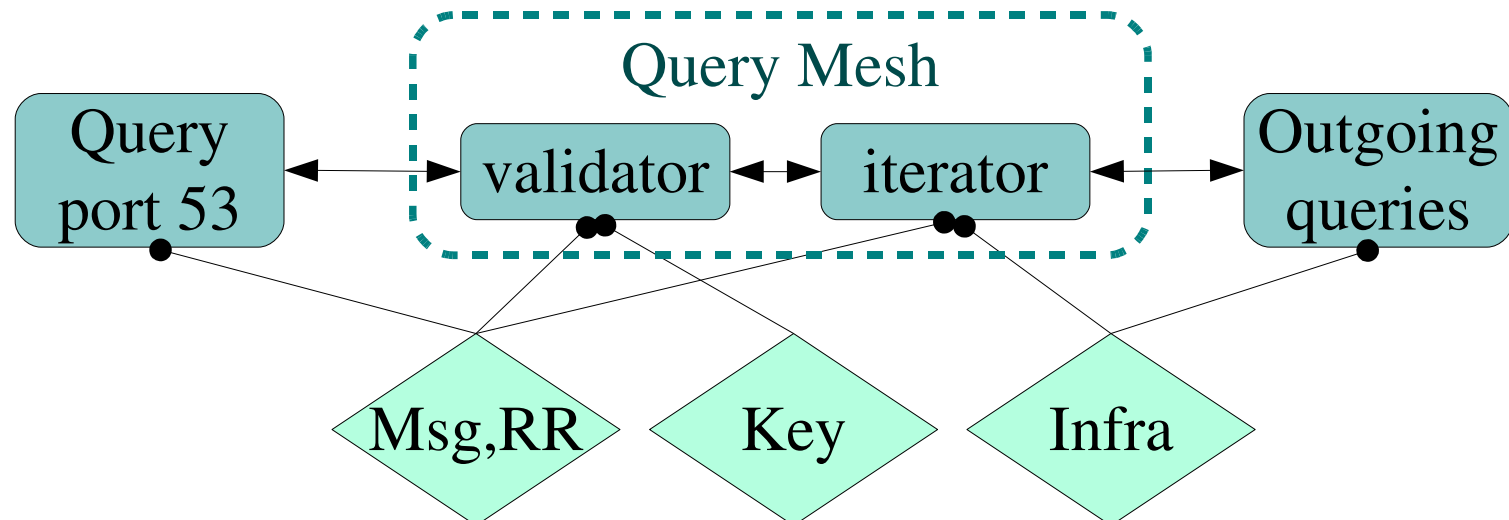
- Trust anchors: *feature rich*
 - Rbtree for anchors – many islands
 - DS and DNSKEY can be used for the anchor
 - Zone-format and bind-config style key syntax
- Authority service: *absent*
 - Localhost and reverse (RFC1918) domains
 - Can block domains
 - Not authoritative server, use stub zones

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

Design

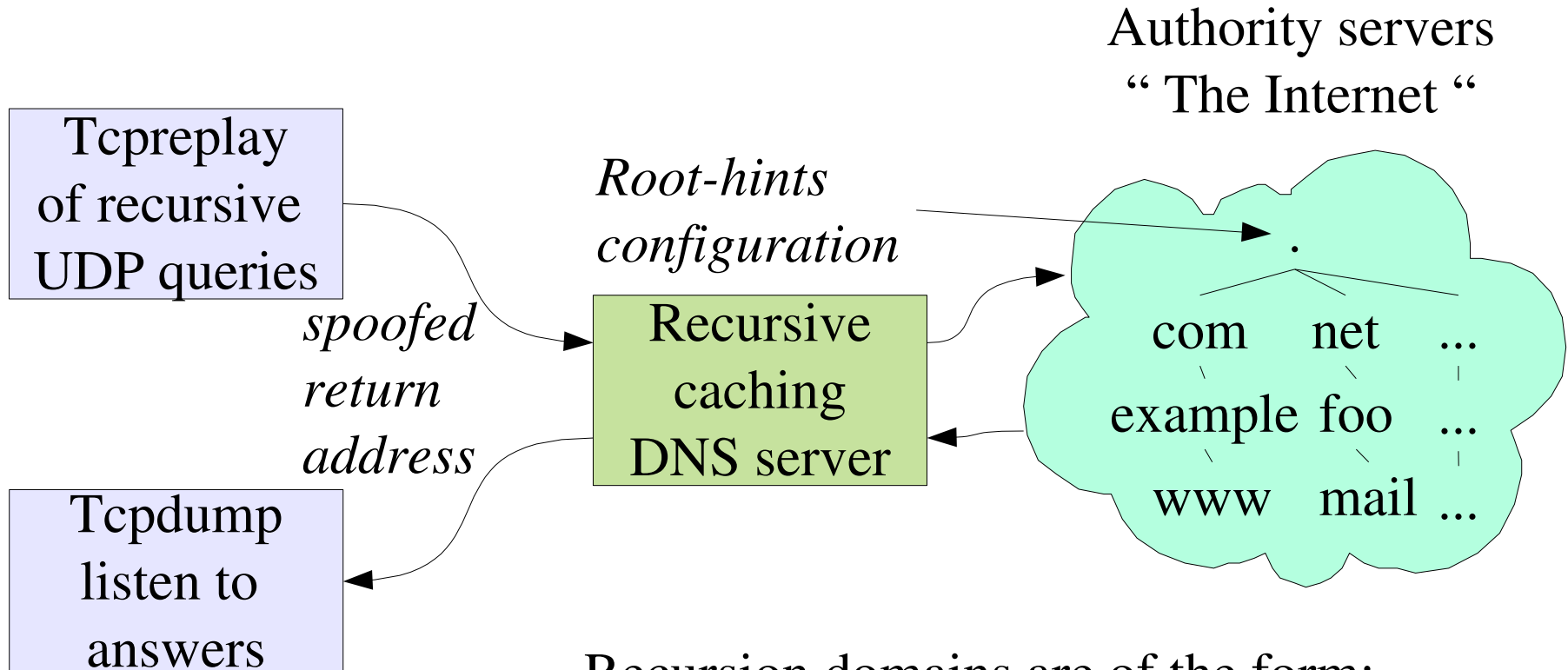
- Worker threads access shared hashtable cache
 - Cache LRU, memory use can be configured
- Modular design, state machines work on query
- Mesh of query dependencies



Tests

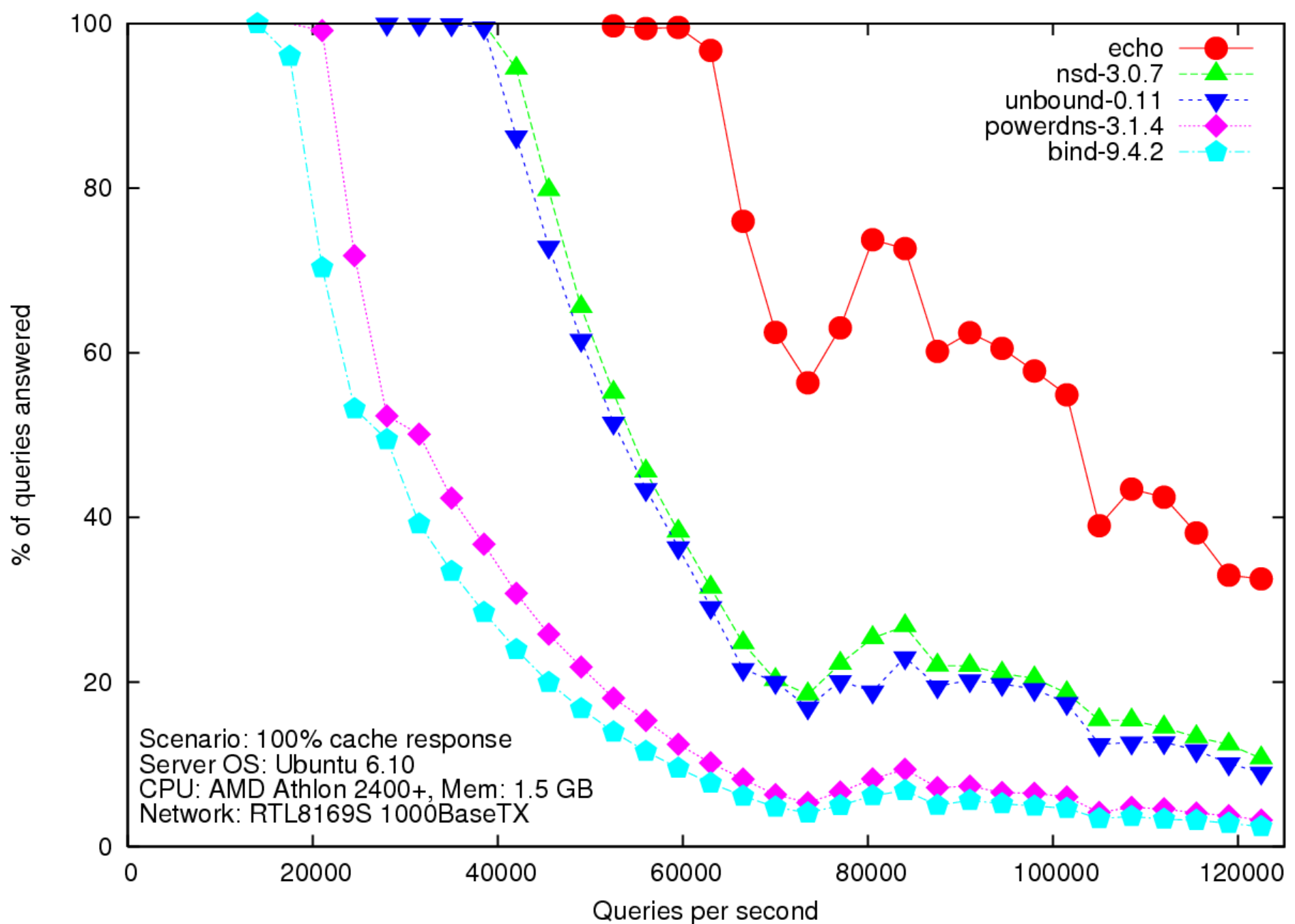
- Regression tests
 - Unit testing of code
 - State machines tested on replay traces
 - Functionality tests (start daemon, make query)
- Beta tests
 - Test in the real world
- Performance tests
 - Cache performance
 - Recursion performance
 - Test against a known, stable environment

Testlab for Resolvers

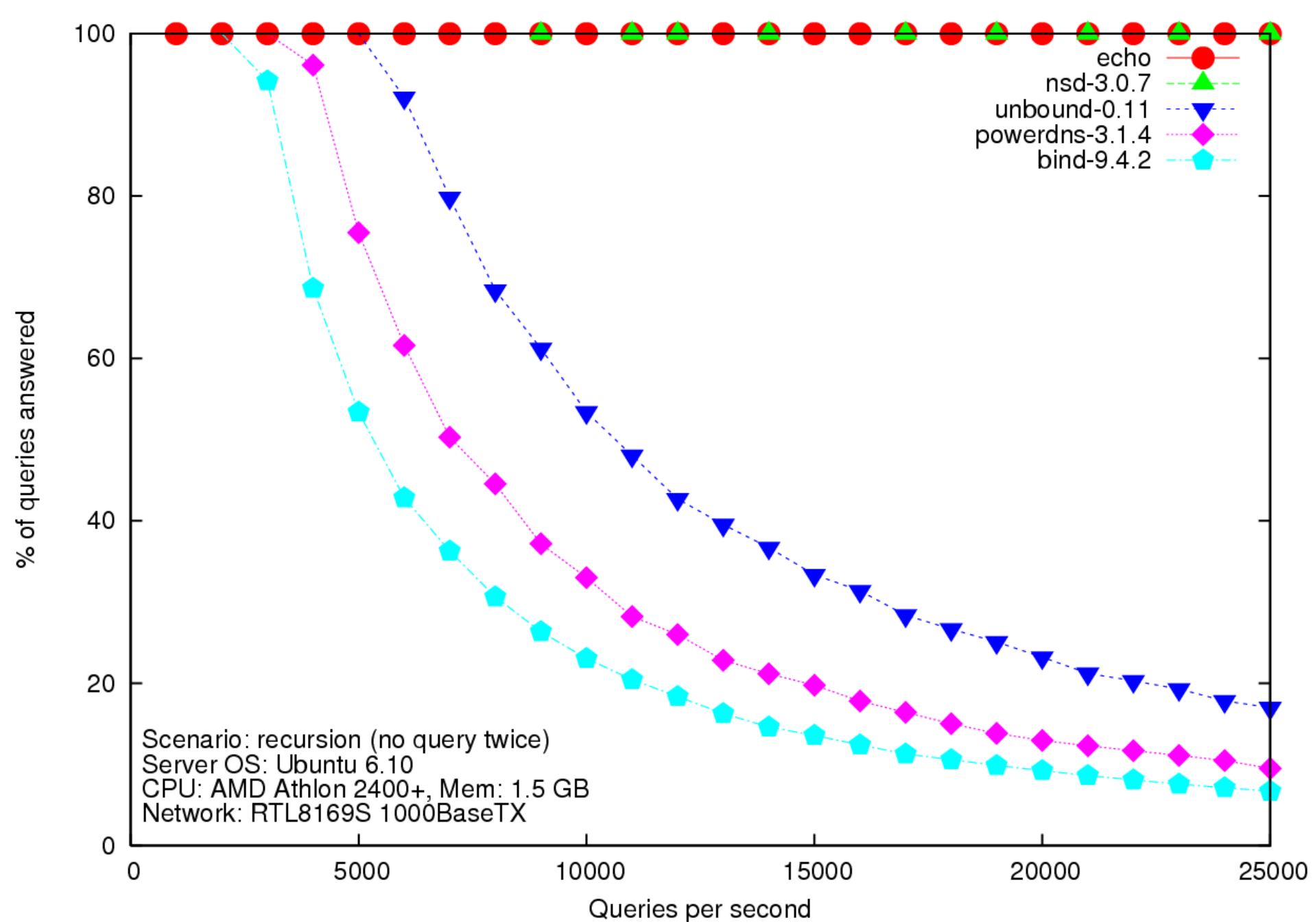


Recursion domains are of the form:

$$\text{www}_{10} \cdot \text{example}_{1000} \cdot \text{com}_{10} \cdot$$

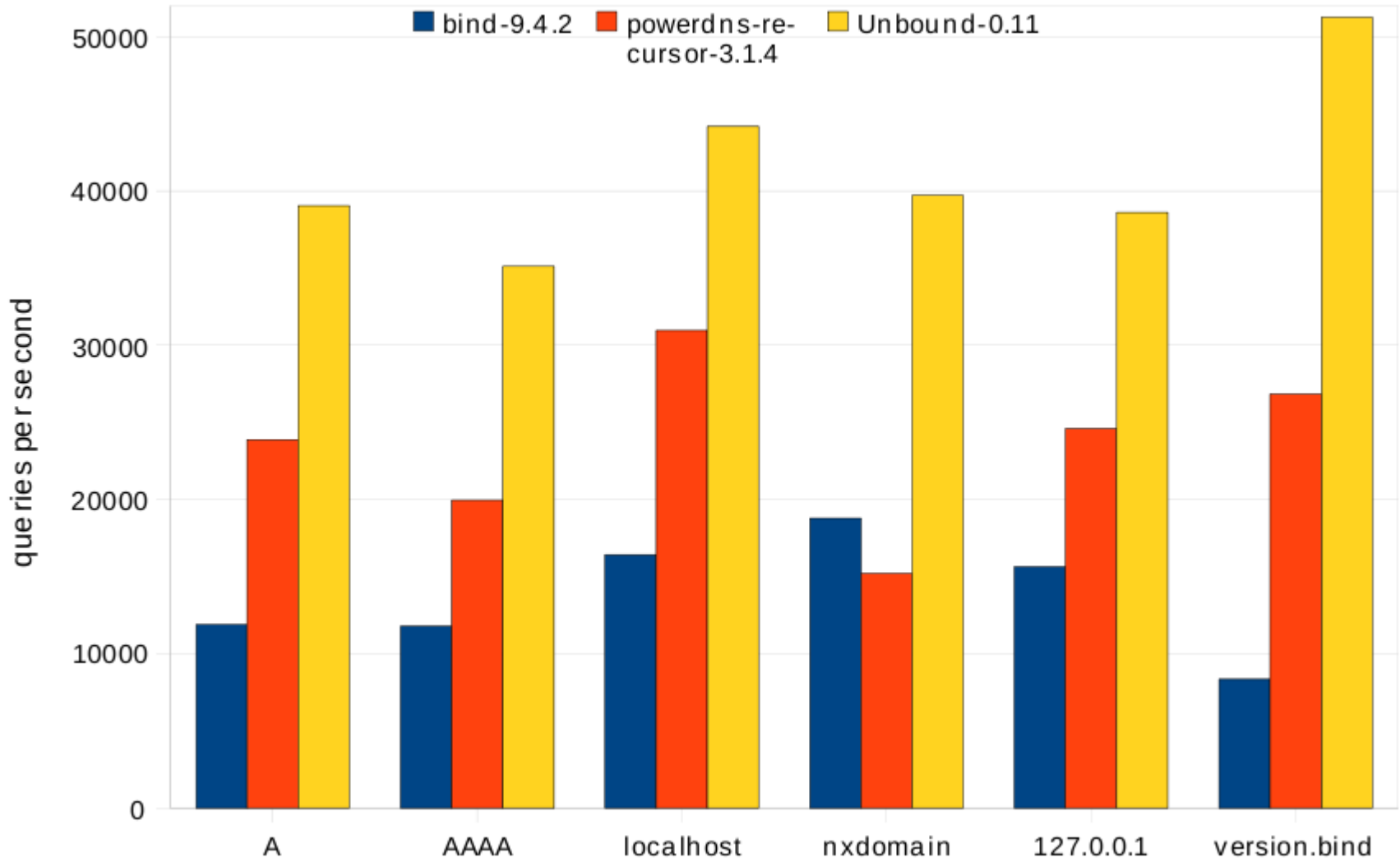


Scenario: 100% cache response
 Server OS: Ubuntu 6.10
 CPU: AMD Athlon 2400+, Mem: 1.5 GB
 Network: RTL8169S 1000BaseTX



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

query perf on test server



Summary

- Unbound – Validating Caching Resolver
 - Open source: BSD license
 - DNSSEC
 - Standards compliant
 - High performance
 - Portable: Linux, *BSD, Solaris, MacOS/X
- Support by NLnet Labs
 - Changes to support announced 2 yrs advance
- Get 0.1.1 at <http://unbound.net>