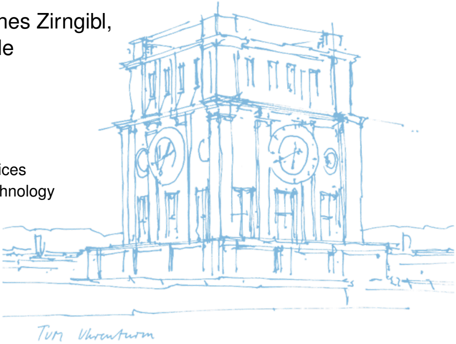
Gotta Query ’Em All, Again!
Repeatable Name Resolution with Full Dependency Provenance

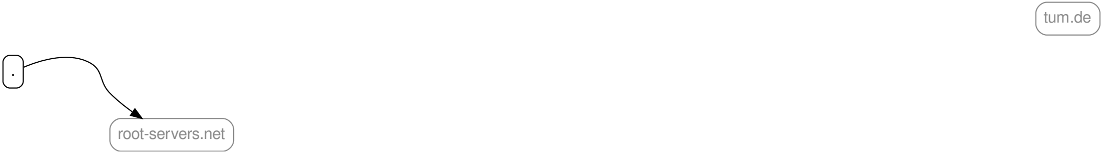
Johannes Naab, Patrick Sattler, Johannes Zirngibl,
Stephan Günther, Georg Carle

Monday 24th July, 2023

Chair of Network Architectures and Services
School of Computation, Information, and Technology
Technical University of Munich

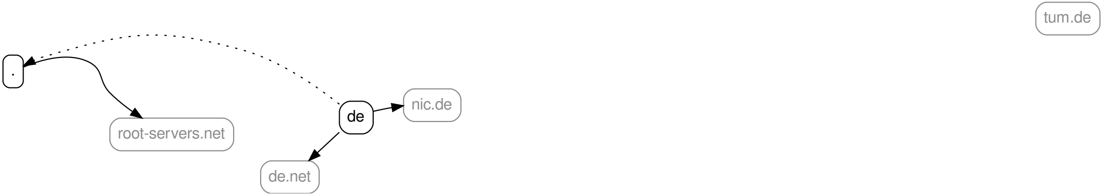


Example: resolving www.tum.de



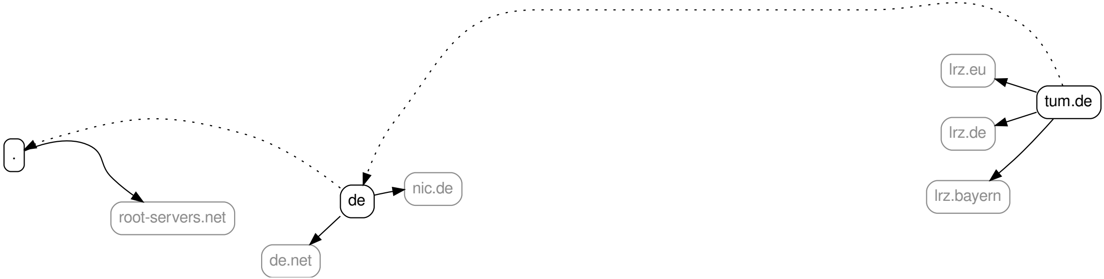
- Starting with the root hints
- Authoritative server IPs are omitted, they are in the zone
- Authoritative server FQDNs are shortened, and point to the enclosing zone (solid arrow)

Example: resolving www.tum.de



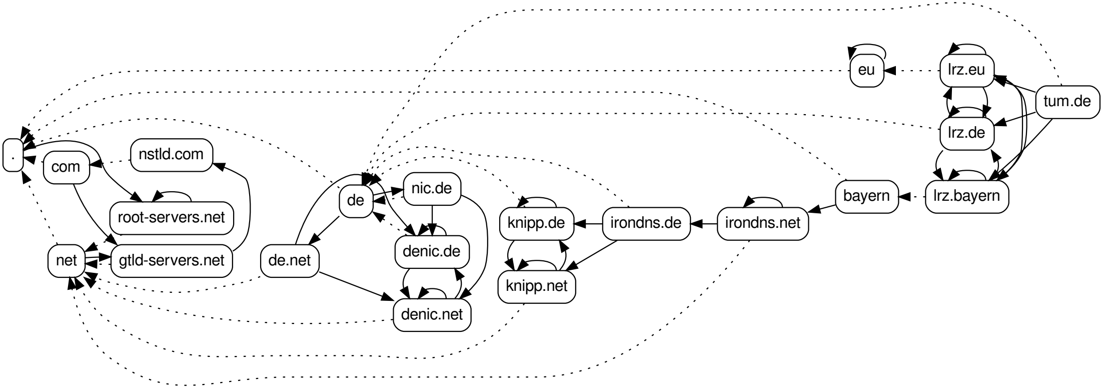
- Referral indication a delegation (dotted arrow pointing from the child to the parent)
- Use glue records as provided in the referrals

Example: resolving www.tum.de



- Delegation for `tum.de` containing glue record for sibling domain `dns1.lrz.de`
- One final query to get the `www.tum.de` answer

Example: Resolving www.tum.de — *With All Dependencies Resolved*



- Follow all resolution paths
- Resolve NS FQDNs within the graph

Motivation

Find and resolve all recursive dependencies

- Construct the entire dependency tree

Identify broken (previously known as lame[1]) delegations

- Authoritative servers that do not exist: NXDomain
- Authoritative servers that do not answer: timeout, ICMP
- Authoritative servers that do not provide useful answers: Refused, ServFail, non-authoritative answer

Multiple resolution paths and data copies

- NS in referral and origin
- Glue records and the authoritative data
- Multiple authoritative servers

[1] for some defintion of

Goals

Research questions

- Study DNS dependency graph
- Find potential inconsistencies and misconfigurations
- Common resolvers do not expose this data
- Build your own resolver: how hard can it be? (-:

Implementation goals

- Discover all reasonable resolution paths
- Query all data copies
- Capture all queries to provide resolution provenance
- Deterministic and repeatable
- Fair and efficient

Implementation

Structure resolution along zones

Find all authoritative server candidates

- glue records and root hints
- resolve NS names within the resolver
- if a parent name server was authoritative

Query `SOA` and `NS` records

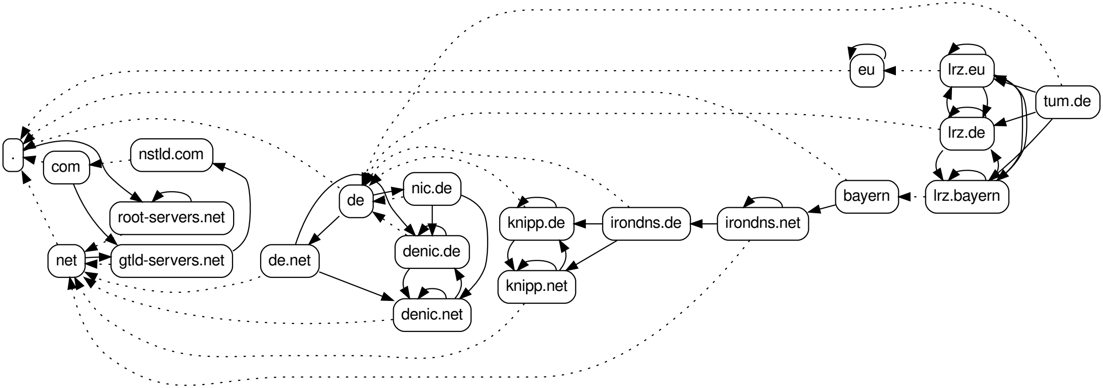
- `NS` record targets need to be resolved
- Name server is considered authoritative if at least one response is authoritative `NoError`

Query all authoritative servers and use the super set of answers

Names found in `NS` records are resolved within the resolver, potentially adding new zones

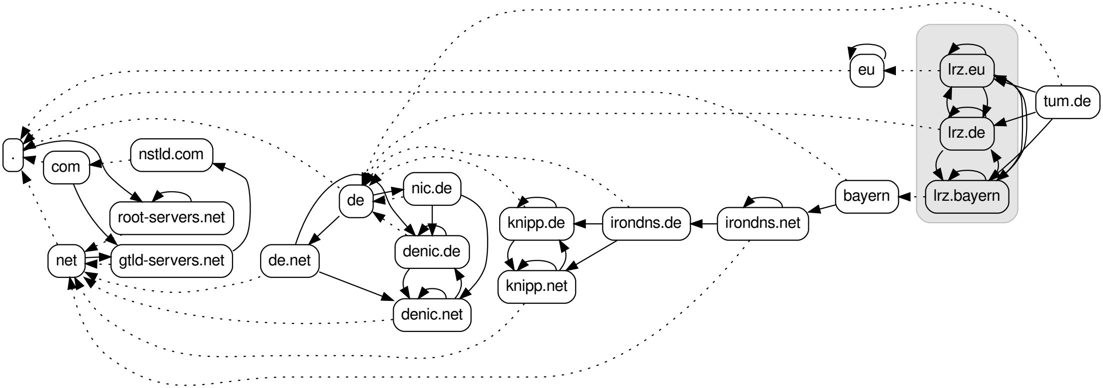
- Is it name resolution all the way down?

Name Resolution All the Way Down



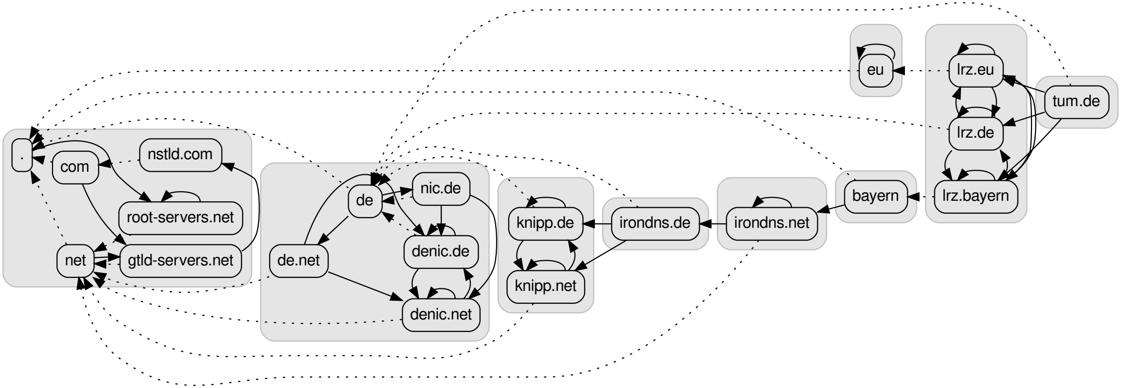
- Groups of interdependent zones
- E. g. `lrz.eu`, `lrz.de`, and `lrz.bayern`

Name Resolution All the Way Down



- Strongly connected component (SCC)
- In DNS: groups of zones where anyone can impact all other including itself

Name Resolution All the Way Down



- Complete resolution along SCCs, starting at the root
- Postpone queries until components are identified . . .
- . . . unless they are found to be necessary for an SCC as identified by only graph search

Zone Cut Detection

SCCs require zone identification

- All dots within a domain are potential zone cuts

Qname minimization provides a frame work

- Query all potential zone cuts
- Use `SOA` queries since `A` queries can hide the zone cut if the parent is authoritative for the child as well

Left most label (e. g. www.tum.de) within an effective second-level domain is likely not a zone cut

- Avoid sending `SOA` query only to discover nothing unless the answers themselves indicate a separate zone

Query Efficiency

Querying all authoritative servers explicitly might not be viable

- 26 authoritative servers in `.com` and a large zone
- The chosen per-name-server rate limit bottlenecks the resolution for all `.com` domains
- Verisign as operator would (likely) prefer not to answer the same query 26 times

Query only a subset of authoritative servers

- Assume TLD servers are consistent and properly managed
- Deterministic subset of 3 authoritative servers based on qname and servers
- Query all servers if any discrepancy is discovered

Additional optimization

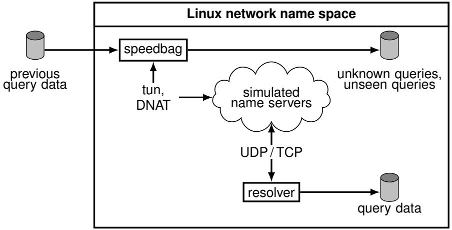
- Inject referrals learned from zone files (CZDS) to avoid them all together
- Not for queries triggered by the resolver itself to be able to rerun the process

Testing

Implementing resolvers involves lots of bugs, trial and error

(Re)running against the Internet

- Queries burden authoritative servers
- Results are not 1:1 comparable due to changes in the DNS



Run against a simulation providing previously recorded responses

- Allows rerunning without involving the authoritative servers
- Record unknown queries (not seen in the original data set)
- Record unseen queries (those in the original data set, but not queried)

Due to timeout handling, compare a certain set of runs with each other to determine repeatability and determinism

Conclusion

Resolver to discover the entire dependency tree

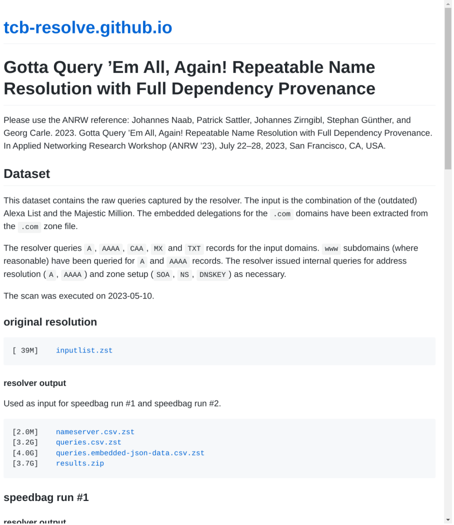
- Repeatable and deterministic name resolution
- Saving all reasonable resolution paths for later analysis
- Process to run tests and ensure repeatability and determinism

Data set sample at tcb-resolve.github.io

- 1.6 M domains from Alexa and Majectic Million lists
- `A`, `AAAA`, `TXT`, `MX`, `CAA`, sensible `www` subdomains `A`, `AAAA`
- 118 M queries, 1.67 M zones, 254 k name servers

Future Work

- Data needs to be analyzed
- Impact of inconsistencies and misconfigurations evaluated
- Open to new / interesting question that could be answered by such data sets



tcb-resolve.github.io

Gotta Query 'Em All, Again! Repeatable Name Resolution with Full Dependency Provenance

Please use the ANRW reference: Johannes Naab, Patrick Sattler, Johannes Zirngibl, Stephan Günther, and Georg Carle. 2023. Gotta Query 'Em All, Again! Repeatable Name Resolution with Full Dependency Provenance. In Applied Networking Research Workshop (ANRW '23), July 22–28, 2023, San Francisco, CA, USA.

Dataset

This dataset contains the raw queries captured by the resolver. The input is the combination of the (outdated) Alexa List and the Majestic Million. The embedded delegations for the `.com` domains have been extracted from the `.com` zone file.

The resolver queries `A`, `AAAA`, `CAA`, `MX` and `TXT` records for the input domains. `www` subdomains (where reasonable) have been queried for `A` and `AAAA` records. The resolver issued internal queries for address resolution (`A`, `AAAA`) and zone setup (`SOA`, `NS`, `DNSKEY`) as necessary.

The scan was executed on 2023-05-10.

original resolution

```
[ 39M]    inputlist.zst
```

resolver output

Used as input for speedbag run #1 and speedbag run #2.

```
[2.0M]    nameserver.csv.zst
[3.2G]    queries.csv.zst
[4.0G]    queries.embedded-json-data.csv.zst
[3.7G]    results.zip
```

speedbag run #1

resolver output