Advanced Computer Networking (ACN)

IN2097 – WiSe 2025–2026

Prof. Dr.-Ing. Georg Carle, Sebastian Gallenmüller

Christian Dietze, Marcel Kempf, Lorenz Lehle

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

Domain Name System

Domain Name System

Domain Name System

Introduction
What is the Domain Name System?

According to RFC 9499 [1], there is no single definition.

The domain name system can be consedered a combination of the following elements [1]:

- a common Internet object naming scheme
- a distributed database (mapping names $\mapsto$ properties of the names, e.g. IPv4 or IPv6 addresses)
- “an architecture providing distributed maintenance, resilience, and loose coherency”
- a rather simple query-response protocol with a bidirectionally consistent message format

Introduction
What is the Domain Name System?

According to RFC 9499 [1], there is no single definition.

The domain name system can be consedered a combination of the following elements [1]:

- a common Internet object naming scheme
- a distributed database (mapping names $\mapsto$ properties of the names, e.g. IPv4 or IPv6 addresses)
- “an architecture providing distributed maintenance, resilience, and loose coherency”
- a rather simple query-response protocol with a bidirectionally consistent message format

It consists of three main components:

1. The domain namespace: a hierarchical namespace with a tree-like structure
2. Nameserver: a program saving information on a part of the namespace
3. Resolver: a program extracting information for clients from the nameservers

- Stub resolver:
  - According to RFC 1034 [2]: Provides recursive resolution for a system which lacks resources (e.g. your PC)

[1] https://www.iana.org/domains/root/servers

Introduction
Terminology

- Stub resolver:
  - According to RFC 1034 [2]: Provides recursive resolution for a system which lacks resources (e.g. your PC)
- Forwarder:
  - Forwards DNS queries to another resolver
  - E.g. your routers resolver

[1] https://www.iana.org/domains/root/servers

Introduction
Terminology

- Stub resolver:
  - According to RFC 1034 [2]: Provides recursive resolution for a system which lacks resources (e.g. your PC)
- Forwarder:
  - Forwards DNS queries to another resolver
  - E.g. your routers resolver
- Recursive resolver
  - Handles recursive queries and iteratively resolves them
  - Usually open resolvers are recursive resolver

1 https://www.iana.org/domains/root/servers

Introduction
Terminology

- Stub resolver:
  - According to RFC 1034 [2]: Provides recursive resolution for a system which lacks resources (e.g. your PC)
- Forwarder:
  - Forwards DNS queries to another resolver
  - E.g. your routers resolver
- Recursive resolver
  - Handles recursive queries and iteratively resolves them
  - Usually open resolvers are recursive resolver
- Authoritative name server
  - Has authoritative information on a set of zones
  - Gets queried by recursive resolvers

1 https://www.iana.org/domains/root/servers

Introduction
Terminology

- Stub resolver:
  - According to RFC 1034 [2]: Provides recursive resolution for a system which lacks resources (e.g. your PC)
- Forwarder:
  - Forwards DNS queries to another resolver
  - E.g. your routers resolver
- Recursive resolver
  - Handles recursive queries and iteratively resolves them
  - Usually open resolvers are recursive resolver
- Authoritative name server
  - Has authoritative information on a set of zones
  - Gets queried by recursive resolvers
- TLD name server
  - Authoritative nameserver for the TLD zones
  - E.g. `a.nic.de` for the `de` zone

1 https://www.iana.org/domains/root/servers

Introduction
Terminology

- Stub resolver:
  - According to RFC 1034 [2]: Provides recursive resolution for a system which lacks resources (e.g. your PC)
- Forwarder:
  - Forwards DNS queries to another resolver
  - E.g. your routers resolver
- Recursive resolver
  - Handles recursive queries and iteratively resolves them
  - Usually open resolvers are recursive resolver
- Authoritative name server
  - Has authoritative information on a set of zones
  - Gets queried by recursive resolvers
- TLD name server
  - Authoritative nameserver for the TLD zones
  - E.g. `a.nic.de` for the `de` zone
- Root server:
  - Authoritative name servers which serve the DNS root zone[1]
  - 13 authorities manage hundreds of servers: [a-m].root-servers.net
  - E.g. `k.root-servers.net` is managed by RIPE
  - `https://root-servers.org/` tracks the location of many root servers

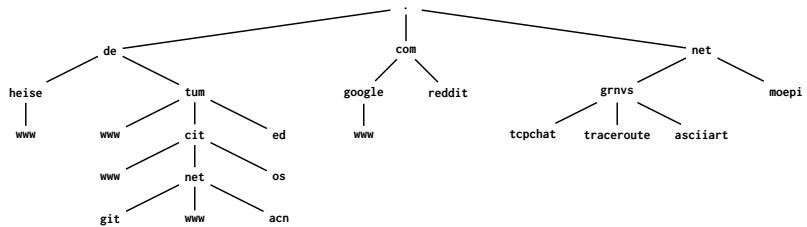
[1] https://www.iana.org/domains/root/servers

Introduction
Hierarchical Structure

- The distributed concept of DNS is based on delegations
- Starting at the root zone a tree of delegations is build

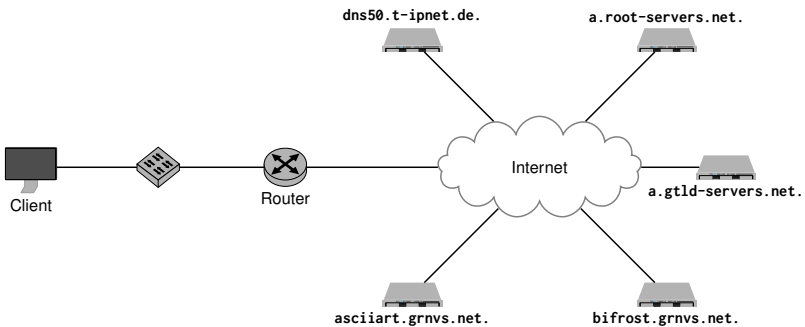
Introduction
Hierarchical Structure

- The distributed concept of DNS is based on delegations
- Starting at the root zone a tree of delegations is build



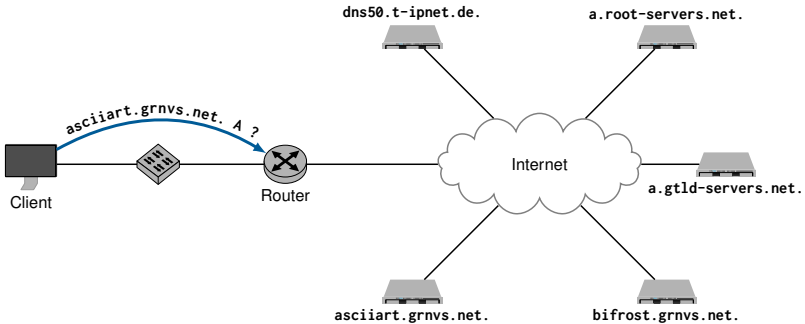
Basic DNS Resolution with a Stub Resolver

- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database

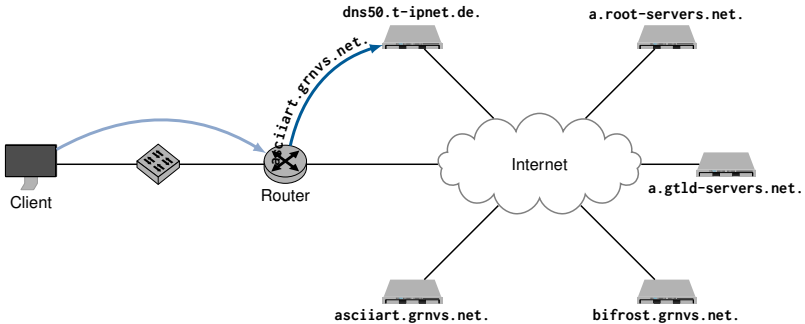


Basic DNS Resolution with a Stub Resolver

- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database

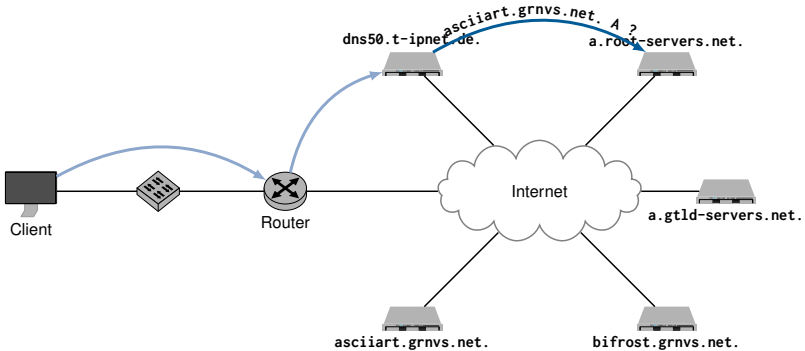


- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database



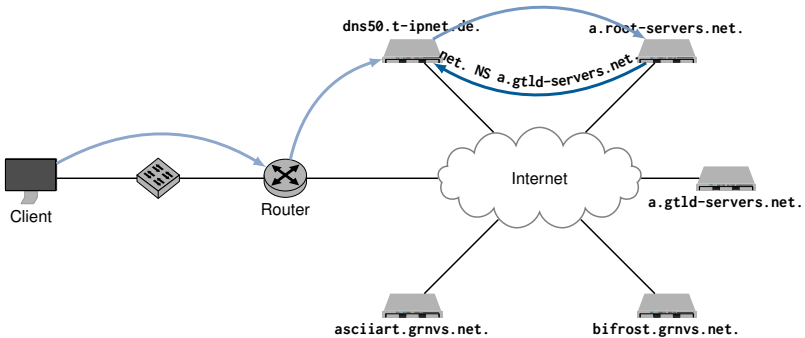
Basic DNS Resolution with a Stub Resolver

- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database



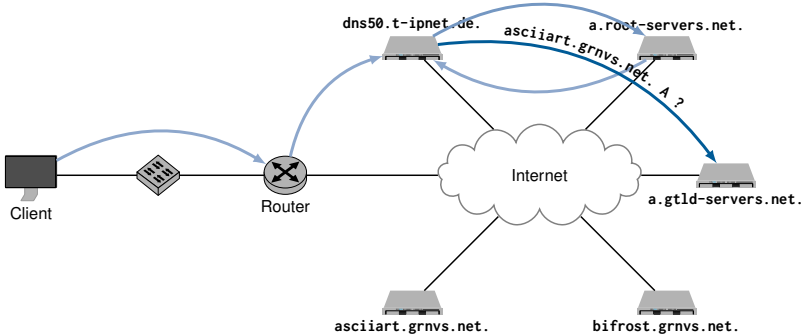
Basic DNS Resolution with a Stub Resolver

- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database



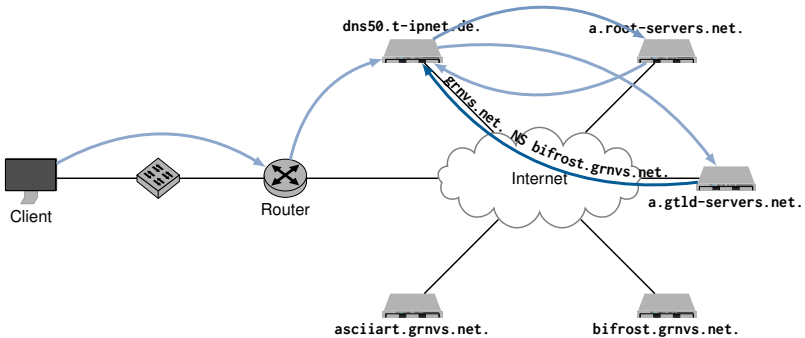
Basic DNS Resolution with a Stub Resolver

- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database



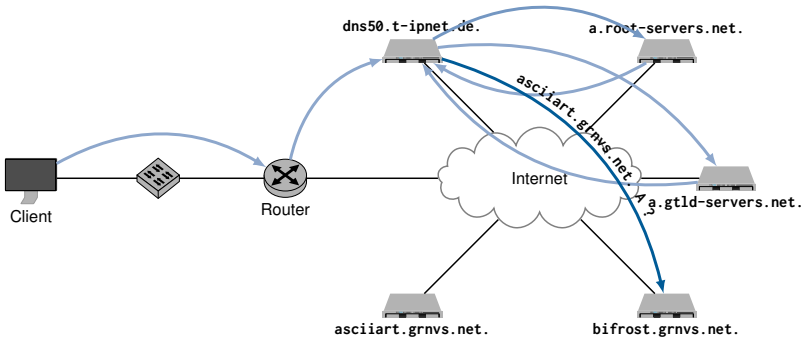
Basic DNS Resolution with a Stub Resolver

- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database



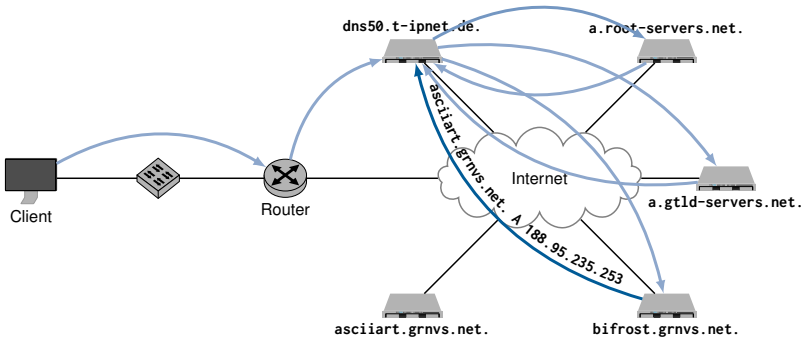
Basic DNS Resolution with a Stub Resolver

- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database

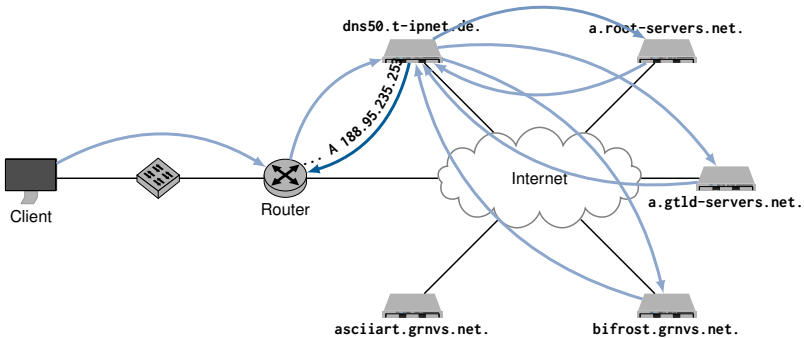


Basic DNS Resolution with a Stub Resolver

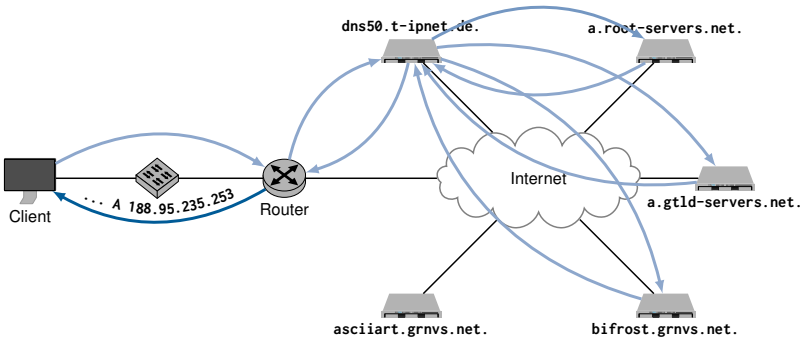
- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database



- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database

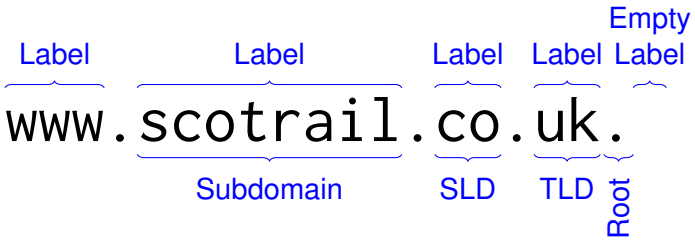


- First standardized in RFC 1034 [2] and 1035 [3]
- System to resolve Fully Qualified Domain Name (FQDN) to IP addresses
- Original concept focused on high scalability → distributed database



Domain Name System

Structure of Domain Names



- Top Level Domain (TLD)
- Second Level Domain (SLD)

[2] List of eTLDs by Mozilla `publicsuffix.org`

Structure of Domain Names

- Top Level Domain (TLD)
- Second Level Domain (SLD)

[2] List of eTLDs by Mozilla `publicsuffix.org`

Structure of Domain Names

- Top Level Domain (TLD)
- Second Level Domain (SLD)

eTLD

- Effective Top Level Domain[2]
- `co.uk`, `com.br`, `gov.br`, ...
- Label before the eTLD: eSLD

[2] List of eTLDs by Mozilla `publicsuffix.org`

Types of Top Level Domains[3]

gTLD: generic Top Level Domain

- A group of TLDs for general purpose domains like `com`, `net`, `org`, `info` or `edu`
- Coordinated by ICANN (see below)

3 https://www.icann.org/en/icann-acronyms-and-terms

Domain Names
Types of Top Level Domains[3]

gTLD: generic Top Level Domain

- A group of TLDs for general purpose domains like `com`, `net`, `org`, `info` or `edu`
- Coordinated by ICANN (see below)

new gTLD

- A subgroup of gTLDs created by the new gTLD program
- Introduced to enable expansion of TLDs
- Multiple application rounds managed by ICANN
- Examples: `shop`, `boehringer`, `music`, ...

3 https://www.icann.org/en/icann-acronyms-and-terms

Domain Names
Types of Top Level Domains[3]

gTLD: generic Top Level Domain

- A group of TLDs for general purpose domains like `com`, `net`, `org`, `info` or `edu`
- Coordinated by ICANN (see below)

new gTLD

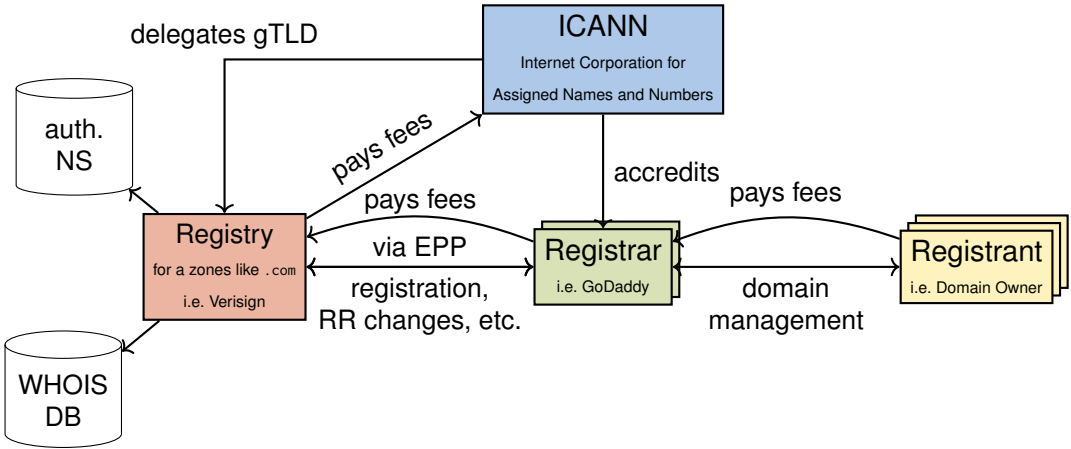
- A subgroup of gTLDs created by the new gTLD program
- Introduced to enable expansion of TLDs
- Multiple application rounds managed by ICANN
- Examples: `shop`, `boehringer`, `music`, ...

ccTLD: country-code Top Level Domain

- TLDs for territories, countries or geographical locations
- General case: two-letter ISO 3166-1 (ALPHA-2) country code
- Exceptions for `ac` (Ascension Island), `uk` (Great Britain and Northern Ireland, `gb`) and `eu` (European Union)

3 https://www.icann.org/en/icann-acronyms-and-terms

ICANN-managed Domains



ICANN-managed Domains II

ICANN: Internet Corporation for Assigned Names and Numbers

- Is a California-based non-profit organisation coordinating unique identifiers used on the Internet
- Tasks:
  - Allocation of TLDs to Registries
  - Coordinates root name server operation
  - Allocation of protocol numbers and identifiers
  - Allocation of IP prefixes to Regional Internet Registries (ICANNs division for this is called IANA)
- Operates with a consensus-based approach

Registry

- Operates authoritative name servers for their TLDs
- Operates the WHOIS database for their TLDs
- Manages domain registrations with the registrars

Registar

- Allows individuals and organisations (registrants) to register and manage a domain name
- Must generally be accredited by the registry (and ICANN for gTLDs)
- Acts as a contact person between the registry and the registrant

Registrant

The entity which owns the domain

Extensible Provisioning Protocol (EPP)

- EPP is “an application-layer client-server protocol for the provisioning and management of objects stored in a shared central repository”[4] defined in RFC 5730
- Based on XML messages following a specific schema
- Used for standardised communication between registry and registrars
- Basic commands include[5]:
  - `login`
  - `logout`
  - `check`
  - `create`
  - `delete`
  - `info`
  - `renew`
  - `transfer`
  - `update`
  - `poll`

4 https://datatracker.ietf.org/doc/html/rfc5730.html

5 Source: https://fred.nic.cz/documentation/html/EPPReference/ProtocolBasics/Commands.html

Extensible Provisioning Protocol (EPP): Example Query[6]

Not Exam Relevant

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:ietf:params:xml:ns:epp
      -1.0_epp-1.0.xsd">
    <command>
        <check>
            <domain:check xmlns:domain="http://www.nic.cz/xml/epp/domain-1.4" xsi:schemaLocation="http://www.nic.cz/xml/epp/domain-1.4_domain
      -1.4.5.xsd">
                <domain:name>
                    available-example.cz
                </domain:name>
                <domain:name>
                    registered-example.cz
                </domain:name>
            </domain:check>
        </check>
        <clTRID>
            nlr23s#2024-04-15T16:07:37.622471
        </clTRID>
    </command>
</epp>
```

6 https://fred.nic.cz/documentation/html/EPPReference/CommandStructure/Check/CheckDomain.html

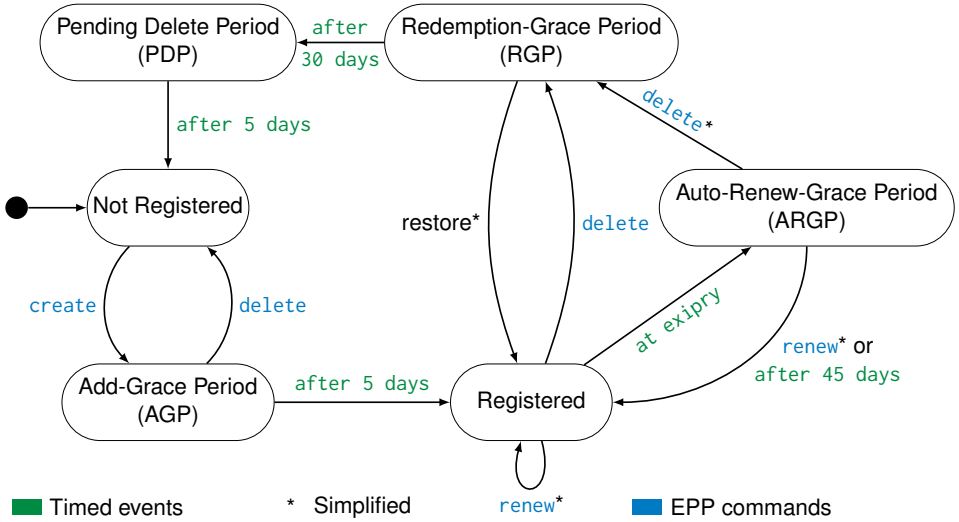
Extensible Provisioning Protocol (EPP): Example Response[7]

Not Exam Relevant

```
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="urn:ietf:params:xml:ns:epp
      -1.0_epp-1.0.xsd">
    <response>
        <result code="1000">
            <msg>Command completed successfully</msg>
        </result>
        <resData>
            <domain:chkData xmlns:domain="http://www.nic.cz/xml/epp/domain-1.4" xsi:schemaLocation="http://www.nic.cz/xml/epp/domain-1.4_domain
      -1.4.5.xsd">
                <domain:cd>
                    <domain:name avail="1">available-example.cz</domain:name>
                </domain:cd>
                <domain:cd>
                    <domain:name avail="0">registered-example.cz</domain:name>
                    <domain:reason>Registered already</domain:reason>
                </domain:cd>
            </domain:chkData>
        </resData>
        <trID>
            <clTRID>nlr23s#2024-04-15T16:07:37.622471</clTRID>
            <svTRID>ReqID-0063282953</svTRID>
        </trID>
    </response>
</epp>
```

7 https://fred.nic.cz/documentation/html/EPPReference/CommandStructure/Check/CheckDomain.html

gTLD Domain Life Cycle from the View of the Registry (adapted from [4])



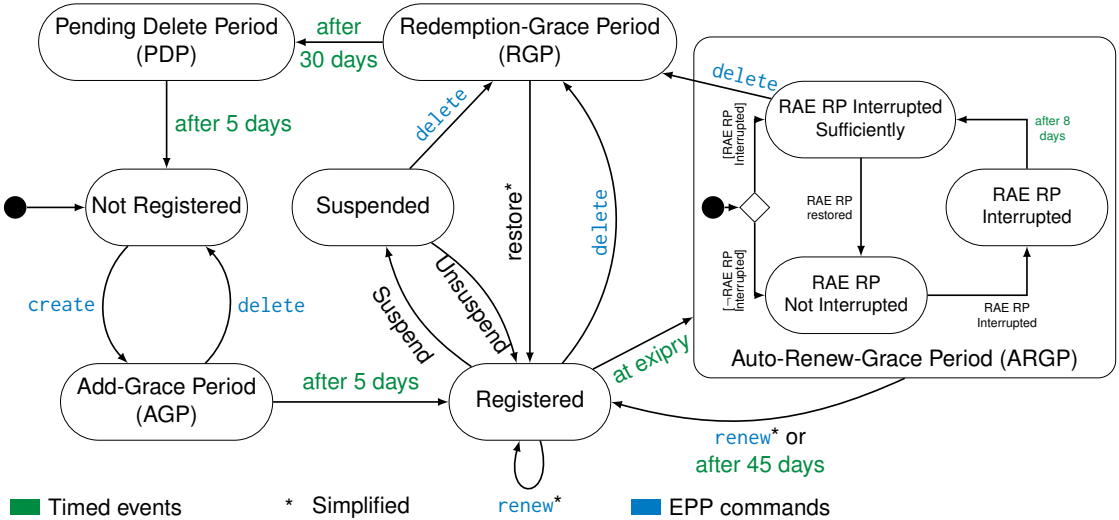
Domain Life Cycle

- Different TLDs have different domain life cycles
- For gTLDs:
  - ICANN contractually fixes many details regarding domain life cycle and conditions in the registry agreement (and their appendixes) between the registry and ICANN
  - Registrars still have certain freedoms, especially regarding the expiration life cycle and costs

  → Check the terms, conditions and reputation of different registrars before choosing one
  - Different gTLD registry agreements can have different conditions, but the live cycle for most is very similar
- For ccTLDs:
  - As each ccTLD registry is independent, their life cycles are vastly different
  - e.g. de has no grace period like the Auto-Renew Grace Period (ARGP)

gTLD Domain Life Cycle from the View of the Registar (adapted from [4] with Information from [5])

Not Exam Relevant



WHOIS

- Protocol to query information on domain names, ASes and delegated prefixes
- Documented in RFC 3192
- TCP-based
- Requests are terminated with `\r\n`
- Responses are terminated by closing the TCP connection
- Information is not consistently formatted and designed to be human-readable

Thick vs Thin Registration: Example

WHOIS response from registry WHOIS server

```
   Domain Name: GRNVS.NET
   Registry Domain ID: 1925142616_DOMAIN_NET-VRSN
   Registrar WHOIS Server: whois.joker.com
   Registrar URL: http://www.joker.com
   Updated Date: 2025-02-25T13:34:23Z
   Creation Date: 2015-05-02T12:37:23Z
   Registry Expiry Date: 2027-05-02T12:37:23Z
   Registrar: CSL Computer Service Langenbach GmbH d/b/a joker.com
   Registrar IANA ID: 113
   Registrar Abuse Contact Email: abuse@joker.com
   Registrar Abuse Contact Phone: +49.21186767447
   Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
   Name Server: BIFROST.GRNVS.NET
   Name Server: SKJOLD.GRNVS.NET
   DNSSEC: unsigned
   URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of whois database: 2025-11-20T15:55:09Z <<<
...
```

Thick vs Thin Registration: Example

WHOIS response from registrar WHOIS server

```
Domain Name: grnvs.net
Registry Domain ID: 1925142616_DOMAIN_NET-VRSN
Registrar WHOIS Server: whois.joker.com
Registrar URL: https://joker.com
Updated Date: 2025-02-25T13:34:23Z
Creation Date: 2015-05-02T12:37:23Z
Registrar Registration Expiration Date: 2027-05-02T12:37:23Z
Registrar: CSL Computer Service Langenbach GmbH d/b/a joker.com
Registrar IANA ID: 113
Registrar Abuse Contact Email: abuse@joker.com
Registrar Abuse Contact Phone: +49.21186767447
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Registrant Organization: Technische Universitaet Muenchen
Registrant Country: DE
Registrant Email: https://csl-registrar.com/contact/grnvs.net/owner
Admin Email: https://csl-registrar.com/contact/grnvs.net/admin
Tech Email: https://csl-registrar.com/contact/grnvs.net/tech
Name Server: bifrost.grnvs.net
Name Server: skjold.grnvs.net
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of WHOIS database: 2025-11-20T15:55:22Z <<<
...
```

Thick vs Thin Registration[8]

Thin Registration

- Registry operator only maintains and provides technical information like nameservers, statuses, creation date and sponsoring registrar
- Registrar associated with the domain provides contact information
- Legacy variant
- Most TLDs/registrations have switched to thick registrations

Thick Registration

- Like thin registration
- Registry operator has a copy of the contact information from the registrars
- The registry operators WHOIS server can already serve all WHOIS information
- Default for all new gTLDs, transition progress for older gTLDs planned

8 Source: https://www.icann.org/en/contracted-parties/consensus-policies/thick-registry-registration-data-directory-services-transition-policy/thick-whois-transition-policy-for-com-net-and-jobs-01-02-2017-en

Registration Data Access Protocol (RDAP)

- Replacement for the WHOIS protocol
- Standardised in RFC 7482
- RESTful API design
- Standardised command, input and output structures
- Example: query domain information for `grnvs.net`: `https://rdap.verisign.com/net/v1/domain/grnvs.net`

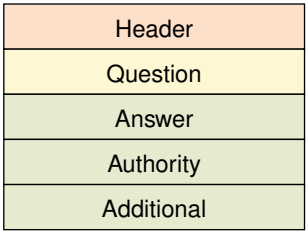
Registration Data Access Protocol (RDAP)

- Replacement for the WHOIS protocol
- Standardised in RFC 7482
- RESTful API design
- Standardised command, input and output structures
- Example: query domain information for `grnvs.net`: `https://rdap.verisign.com/net/v1/domain/grnvs.net`


```
{
  "objectClassName": "domain",
  "handle": "1925142616_DOMAIN_NET-VRSN",
  "ldhName": "GRNVS.NET",
  "links":[{"value":"https://rdap.verisign.com/net/v1/domain/GRNVS.NET","rel":"self","href":"https://rdap.verisign.com/net/v1/domain/GRNVS.NET",
            "type":"application/rdap+json"},
           {"value":"https://rdap.joker.com/domain/GRNVS.NET","rel":"related","href":"https://rdap.joker.com/domain/GRNVS.NET","type":"application/rdap+json"}],
  "status": ["client transfer prohibited"],
  "entities":[{"objectClassName":"entity","handle":"113","roles":["registrar"],
               "links":[{"href":"http://www.joker.com","type":"text/html","value":"https://rdap.joker.com/","rel":"about"}],
               "publicIds":[{"type":"IANA Registrar ID","identifier":"113"}],
               "vcardArray":["vcard",[["version",{},"text","4.0"],["fn",{},"text","CSL Computer Service Langenbach GmbH d/b/a joker.com"]]],
               "entities":[{"objectClassName":"entity","roles":["abuse"],"vcardArray":["vcard",[["version",{},"text","4.0"],["fn",{},"text",""],
                   ["tel",{"type":"voice"},"uri","tel:+49.21186767447"],["email",{},"text","abuse@joker.com"]]]}]}],
  "events": [{"eventAction": "registration", "eventDate": "2015-05-02T12:37:23Z"}, {"eventAction": "expiration", "eventDate": "2027-05-02T12:37:23Z"},
    {"eventAction": "last changed" ,"eventDate": "2025-02-25T13:34:23Z"}, {"eventAction": "last update of RDAP database", "eventDate": "2025-11-21T11:06:09Z"}],
  "secureDNS": {"delegationSigned": false},
  "nameservers": [{"objectClassName": "nameserver", "ldhName": "BIFROST.GRNVS.NET"}, {"objectClassName": "nameserver", "ldhName": "SKJOLD.GRNVS.NET"}],
  "rdapConformance": ["rdap_level_0", "icann_rdap_technical_implementation_guide_1", "icann_rdap_response_profile_1"],
  "notices": [...]
}
```

Domain Name System

DNS Message Format
General Message Format



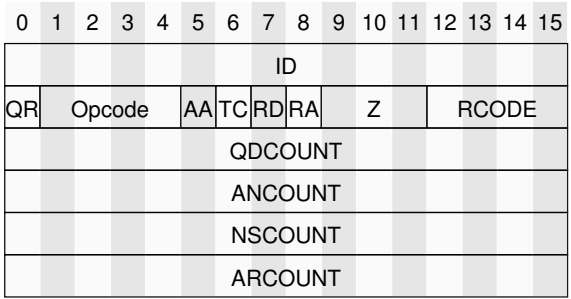
The question of the name server

RRs answering the question

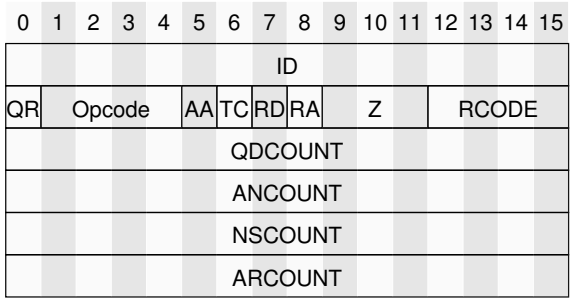
RRs pointing toward an authority

RRs holding additional information

- Query and response use same message format
- Header indicates type of message
- The question section contains the requested entry
- The answer, authority, and additional section are *arrays* of resource records (RR)

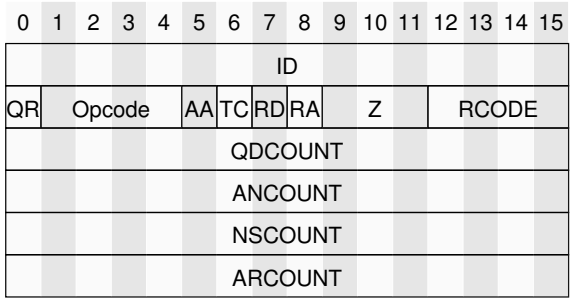


9 https://www.iana.org/assignments/dns-parameters contains all defined opcode an rcode values



ID Unique query ID to identify the corresponding response

9 https://www.iana.org/assignments/dns-parameters contains all defined opcode an rcode values

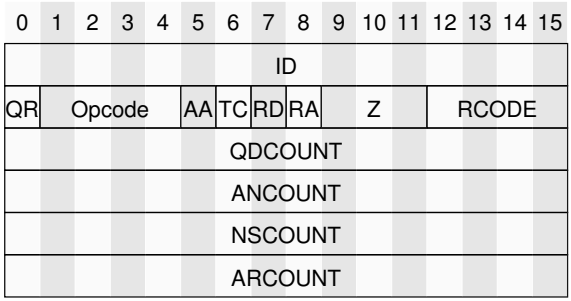


ID Unique query ID to identify the corresponding response

QR Set if the message is a response

9 https://www.iana.org/assignments/dns-parameters contains all defined opcode an rcode values

Format I



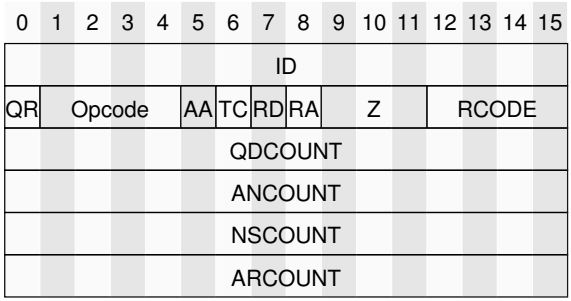
ID Unique query ID to identify the corresponding response

QR Set if the message is a response

Opcode Specifies kind of query (e.g. query, status, notify, update)[9]

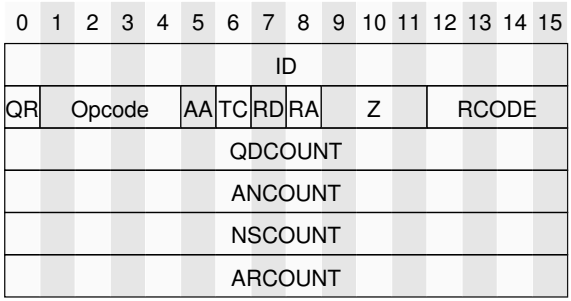
9 `https://www.iana.org/assignments/dns-parameters` contains all defined opcode an rcode values

Format I



ID	Unique query ID to identify the corresponding response
QR	Set if the message is a response
Opcode	Specifies kind of query (e.g. query, status, notify, update)[9]
AA: Authoritative Answer	Set if the responding name server is an authority for the requested domain

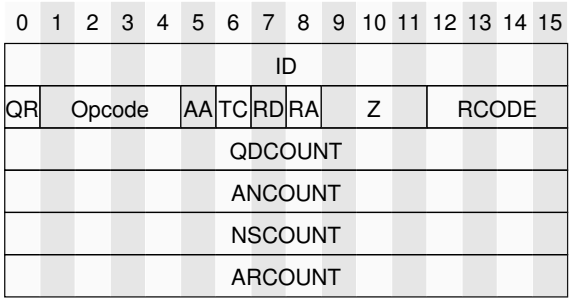
9 `https://www.iana.org/assignments/dns-parameters` contains all defined opcode an rcode values



ID	Unique query ID to identify the corresponding response
QR	Set if the message is a response
Opcode	Specifies kind of query (e.g. query, status, notify, update)[9]
AA: Authoritative Answer	Set if the responding name server is an authority for the requested domain
TC: Truncated	Indicates that the DNS message is truncated due to the permitted length

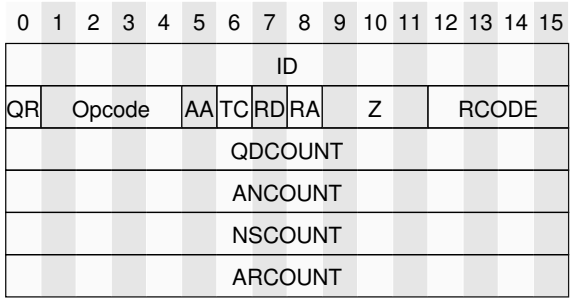
9 `https://www.iana.org/assignments/dns-parameters` contains all defined opcode an rcode values

Format II



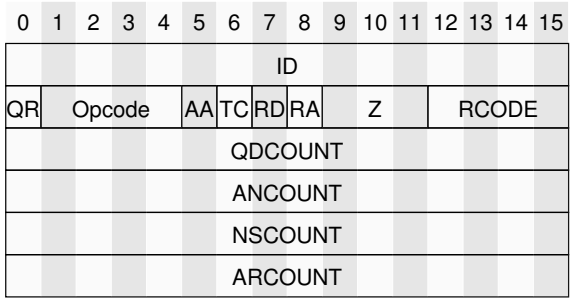
RD: Recursion desired If set the nameserver resolves the query recursively

Format II



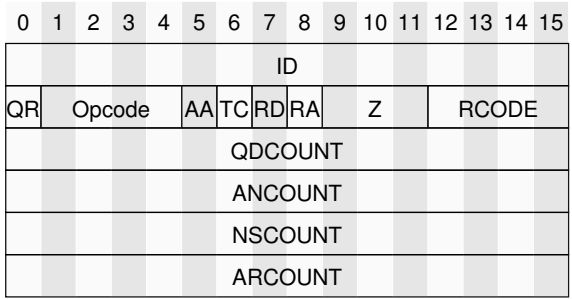
RD: Recursion desired	If set the nameserver resolves the query recursively
RA: Recursion available	Set by the nameserver if it supports recursive queries

Format II

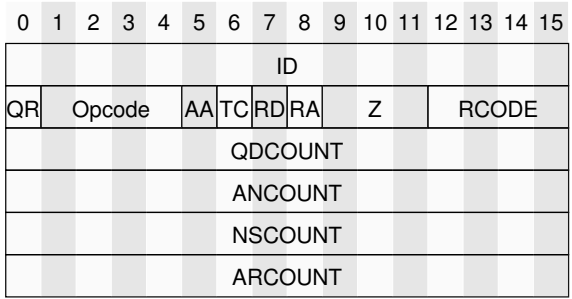


RD: Recursion desired	If set the nameserver resolves the query recursively
RA: Recursion available	Set by the nameserver if it supports recursive queries
Z	1 bit future use; 2 bits for DNSSEC (authentic data (AD) and checking disabled (CD))

Format II

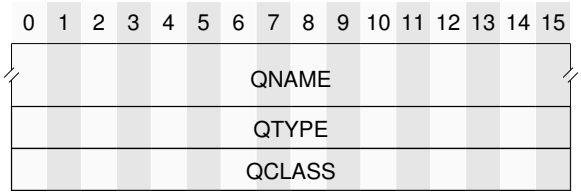


RD: Recursion desired	If set the nameserver resolves the query recursively
RA: Recursion available	Set by the nameserver if it supports recursive queries
Z	1 bit future use; 2 bits for DNSSEC (authentic data (AD) and checking disabled (CD))
RCODE: Response code	Code indicating query status (e.g. NOERROR, NXDOMAIN, SERVFAIL)

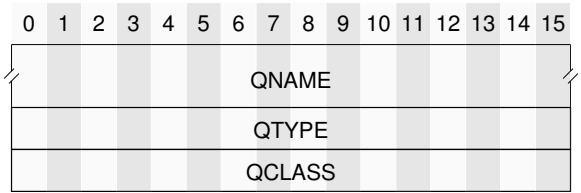


RD: Recursion desired	If set the nameserver resolves the query recursively
RA: Recursion available	Set by the nameserver if it supports recursive queries
Z	1 bit future use; 2 bits for DNSSEC (authentic data (AD) and checking disabled (CD))
RCODE: Response code	Code indicating query status (e.g. NOERROR, NXDOMAIN, SERVFAIL)
*COUNT	Number of RR in the corresponding message section

Question Section
Format

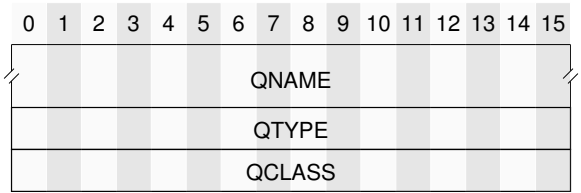


Question Section
Format



QNAME Requested Name, variable length

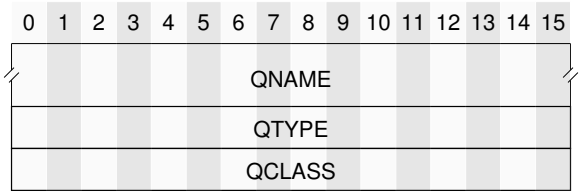
Question Section
Format



QNAME Requested Name, variable length

QTYPE Requested RR Type (e.g., A, NS)

Question Section
Format

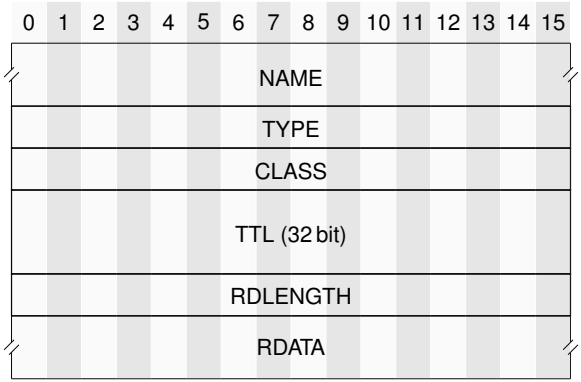


QNAME Requested Name, variable length

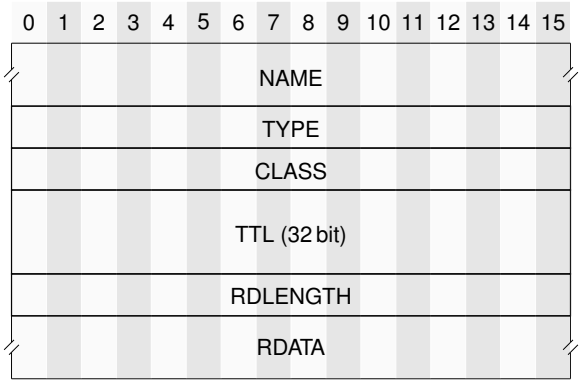
QTYPE Requested RR Type (e.g., A, NS)

QCLASS Normally Internet (IN)

Format of a Single Resource Record

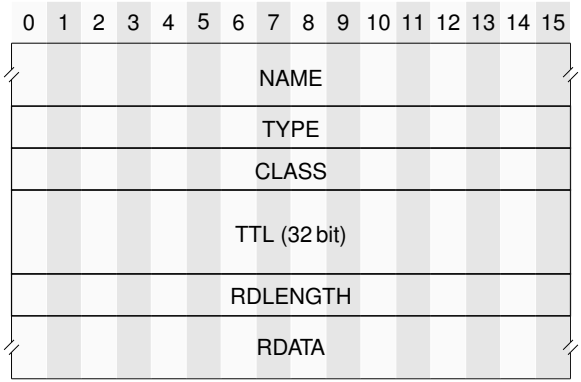


Format of a Single Resource Record



TTL Valid lifetime of the RR in seconds

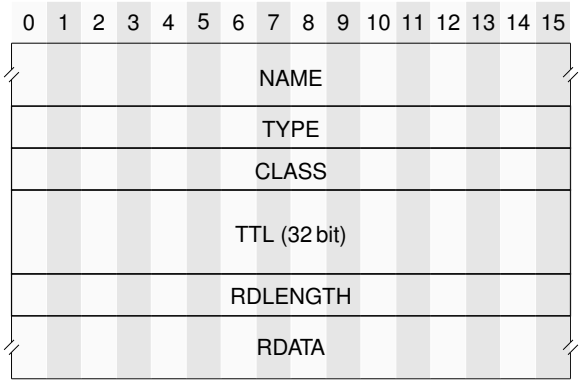
Format of a Single Resource Record



TTL Valid lifetime of the RR in seconds

RDLENGTH Length of the following data

Format of a Single Resource Record

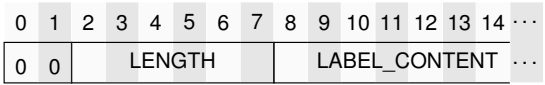


TTL	Valid lifetime of the RR in seconds
RDLENGTH	Length of the following data
RDATA	Data of the RR mapped to the name

Binary Format of Domain Names

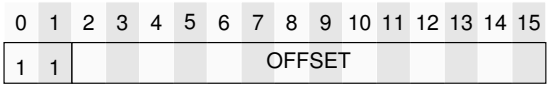
- Defined in RFC 1035
- A domain name consists of a sequence of labels
- There are two types of labels:
  1. (Ordinary) Labels
  2. Pointer
- The sequence is terminated by a label of length 0 representing the root

Labels



- One byte length field followed by that many bytes of label content
- The first two bits of the length field are always `0b00`
- Effective maximal label content size: $2^{8-2} - 1 = 64 - 1 = 63$

Pointers



- Used to reduce the DNS message size (message compression) by eliminating repetition
- The first two bits being `0b11`
- The rest of the pointer's two bytes (called offset) is an offset from the DNS message start
- The label sequence continues at the offset

Answer, Authority and Additional Sections

Select Type Values

Type	Meaning	Representation
A	an IPv4 host address	32 bit address
AAAA	an IPv6 host address	128 bit address

10 https://datatracker.ietf.org/doc/draft-ietf-dnsop-svcb-https/

Answer, Authority and Additional Sections
Select Type Values

Type	Meaning	Representation
A	an IPv4 host address	32 bit address
AAAA	an IPv6 host address	128 bit address
CNAME	canonical name for an alias	a domain name
NS	autorithative name server	domain name
SOA	start of zone authority	Various fields

SOA contains:

- domain name of primary name server
- domain name of mailbox of responsible person
- serial 32 bit value
- refresh 32 bit value = ttl of zone
- retry 32 bit value
- expire 32 bit value
- minimum 32 bit value of minimum ttl value in zone. negative caching ttl

[10] https://datatracker.ietf.org/doc/draft-ietf-dnsop-svcb-https/

Select Type Values

Type	Meaning	Representation
A	an IPv4 host address	32 bit address
AAAA	an IPv6 host address	128 bit address
CNAME	canonical name for an alias	a domain name
NS	autorithative name server	domain name
SOA	start of zone authority	Various fields
MX	Mail exchange address	Preference (16-bit, lower are preferred) and mail server domain name
TXT	TXT record	Arbitrary text (character-strings)
PTR	points to a domain name	domain name

SOA contains:

- domain name of primary name server
- domain name of mailbox of responsible person
- serial 32 bit value
- refresh 32 bit value = ttl of zone
- retry 32 bit value
- expire 32 bit value
- minimum 32 bit value of minimum ttl value in zone. negative caching ttl

Character-String: 1 byte `length` + `length` bytes

10 https://datatracker.ietf.org/doc/draft-ietf-dnsop-svcb-https/

Select Type Values

Type	Meaning	Representation
`A`	an IPv4 host address	32 bit address
`AAAA`	an IPv6 host address	128 bit address
`CNAME`	canonical name for an alias	a domain name
`NS`	autorithative name server	domain name
`SOA`	start of zone authority	Various fields
`MX`	Mail exchange address	Preference (16-bit, lower are preferred) and mail server domain name
`TXT`	TXT record	Arbitrary text (character-strings)
`PTR`	points to a domain name	domain name
`SVCB`	service binding record	Information on services[10]
`HTTPS`	HTTPS service binding record	Information on the HTTPS service

`SOA` contains:

- domain name of primary name server
- domain name of mailbox of responsible person
- serial 32 bit value
- refresh 32 bit value = ttl of zone
- retry 32 bit value
- expire 32 bit value
- minimum 32 bit value of minimum ttl value in zone. negative caching ttl

Character-String: 1 byte `length` + `length` bytes

[10] https://datatracker.ietf.org/doc/draft-ietf-dnsop-svcb-https/

Domain Name System

Zones and Delegations
Zone

- RFC 8499 defines a zone:

  Authoritative information is organized into units called ZONEs, and these zones can be automatically distributed to the name servers which provide redundant service for the data in a zone.

- Has a set of name server records (authoritative nameserver)
- Starts with a SOA record, ends at the next SOA record
- Child zone:

  The entity on record that has the delegation of the domain from the Parent.

- Parent zone:

  The domain in which the Child is registered.

- Delegation:

  The process by which a separate zone is created in the name space beneath the apex of a given domain.

Parent zone:

- The zone of the domain name excluding the last label (except ENTs)
- E.g. `de` for `tum.de` and `in.tum.de` for `net.in.tum.de`
- `acn.net.in.tum.de` has no SOA record. I.e. it is not in the zone apex

Parent zone:

- The zone of the domain name excluding the last label (except ENTs)
- E.g. `de` for `tum.de` and `in.tum.de` for `net.in.tum.de`
- `acn.net.in.tum.de` has no SOA record. I.e. it is not in the zone apex
  → it is part of `net.in.tum.de`

Zones and Delegations
Delegations

Parent zone:

- The zone of the domain name excluding the last label (except ENTs)
- E.g. `de` for `tum.de` and `in.tum.de` for `net.in.tum.de`
- `acn.net.in.tum.de` has no SOA record. I.e. it is not in the zone apex
  → it is part of `net.in.tum.de`

Delegations:

- The parent zone has the NS records which delegate the query to the authoritative name server of a zone
- Recap:
  - NS record points to a domain name
  - Either a domain name in the same zone – called in-bailiwick
    - E.g. `ns1.google.com` for `google.com`
  - Or any other domain name (e.g. `dns1.lrz.de` for `net.in.tum.de`)

Zones and Delegations
Delegations

Parent zone:

- The zone of the domain name excluding the last label (except ENTs)
- E.g. `de` for `tum.de` and `in.tum.de` for `net.in.tum.de`
- `acn.net.in.tum.de` has no SOA record. I.e. it is not in the zone apex
  → it is part of `net.in.tum.de`

Delegations:

- The parent zone has the NS records which delegate the query to the authoritative name server of a zone
- Recap:
  - NS record points to a domain name
  - Either a domain name in the same zone – called in-bailiwick
    - E.g. `ns1.google.com` for `google.com`
  - Or any other domain name (e.g. `dns1.lrz.de` for `net.in.tum.de`)
    → Resolver needs to query A/AAAA record of name server name

Parent zone:

- The zone of the domain name excluding the last label (except ENTs)
- E.g. `de` for `tum.de` and `in.tum.de` for `net.in.tum.de`
- `acn.net.in.tum.de` has no SOA record. I.e. it is not in the zone apex
  → it is part of `net.in.tum.de`

Delegations:

- The parent zone has the NS records which delegate the query to the authoritative name server of a zone
- Recap:
  - NS record points to a domain name
  - Either a domain name in the same zone – called in-bailiwick
    - E.g. `ns1.google.com` for `google.com`
  - Or any other domain name (e.g. `dns1.lrz.de` for `net.in.tum.de`)
    → Resolver needs to query A/AAAA record of name server name
  - Problem: How can in-bailiwick records (e.g. of `ns1.google.com`) be retrieved?

Glue Records

- An A/AAAA record in the parent zone for the name server name of a child zone
- Glue records are non authoritative records in the parent zone

Zones and Delegations

```
$TTL 86400 ; 1 day
grnvs.net. IN  SOA    bifrost.grnvs.net. hostmaster.grnvs.net. (
                          164160 ; serial
                          1800   ; refresh (30 minutes)
                          300    ; retry (5 minutes)
                          604800 ; expire (1 week)
                          1800   ; nxdomain (30 minutes)
                      )
               NS     bifrost.grnvs.net.
               NS     forseti.grnvs.net.
               A      131.159.15.21

$ORIGIN grnvs.net.
bifrost        A       129.187.145.241
forseti        A       78.47.25.36
               AAAA    2a01:4f8:190:60a3::2

$TTL 3600 ; 1 hour
traceroute     A       89.163.225.145
               AAAA    2001:4ba0:ffec:0193::0
tcpchat        A       89.163.225.145
asciiart       CNAME   svm0012.net.in.tum.de.

mgmt.testbed   A       138.246.253.25
mgmt.testbed   AAAA    2001:4ca0:108:42::25

testbed        NS      mgmt.testbed
```

Zones and Delegations
Reverse DNS

- It is also possible to query FQDNs belonging to IP addresses using `PTR` records.
- This is called reverse DNS.
- For both IP versions, there are distinct zones:
  - in-addr.arpa. for IPv4
  - ip6.arpa. for IPv6

Zones and Delegations
Reverse DNS

- It is also possible to query FQDNs belonging to IP addresses using `PTR` records.
- This is called reverse DNS.
- For both IP versions, there are distinct zones:
  - in-addr.arpa. for IPv4
  - ip6.arpa. for IPv6
- Example:
  - IPv4: 138.246.253.31

⟶ `31.253.246.138.in-addr.arpa. IN PTR planetlab31.gino-research.net.in.tum.de.`

  - IPv6: `2001:4ca0:108:42::31`

⟶ `1.3.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2.4.0.0.8.0.1.0.0.a.c.4.1.0.0.2.ip6.arpa. IN PTR planetlab31.gino-research.net.in.tum.de.`

Zones and Delegations
Reverse DNS

- It is also possible to query FQDNs belonging to IP addresses using `PTR` records.
- This is called reverse DNS.
- For both IP versions, there are distinct zones:
  - in-addr.arpa. for IPv4
  - ip6.arpa. for IPv6
- Example:
  - IPv4: 138.246.253.31

  ⟶ `31.253.246.138.in-addr.arpa. IN PTR planetlab31.gino-research.net.in.tum.de.`
  - IPv6: `2001:4ca0:108:42::31`

  ⟶ `1.3.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2.4.0.0.8.0.1.0.0.a.c.4.1.0.0.2.ip6.arpa. IN PTR planetlab31.gino-research.net.in.tum.de.`
- RIRs delegate the corresponding zones to the name servers of the operator which had the corresponding prefix delegated.
  - Delegated prefix: `2001:4ca0:108::/48`

  ⟶ `8.0.1.0.0.a.c.4.1.0.0.2.ip6.arpa. 361 IN NS dns1.net.in.tum.de.`
- For reverse resolutions to work, the corresponding `PTR` records have to be created manually.

Zones and Delegations
Empty Non Terminals (ENT)

- Nodes with children but no RRs of their own (RFC2136 Section 7.16)
- Queries for ENTs return NOERROR but n RR in the answer section
- This behavior is important for QNAME minimization and rDNS walking
- E.g.
  - `www.ent.example.com` contains a SOA and an A record
  - `ent.example.com` contains no record
  - `example.com` contains at least a SOA record
  - `ent.example.com` is an ENT

Zones and Delegations
Delegations Issues

- NS record points to an IP address

11 The .io Error – Taking Control of All .io Domains With a Targeted Registration https://thehackerblog.com/the-io-error-taking-control-of-all-io-domains-with-a-targeted-registration/

Zones and Delegations
Delegations Issues

- NS record points to an IP address
  - Not reachable and not valid
  - Reliabilty issue (e.g. other name server are not reachable/overloaded)

11 The .io Error – Taking Control of All .io Domains With a Targeted Registration https://thehackerblog.com/the-io-error-taking-control-of-all-io-domains-with-a-targeted-registration/

Zones and Delegations
Delegations Issues

- NS record points to an IP address
  - Not reachable and not valid
  - Reliabilty issue (e.g. other name server are not reachable/overloaded)

- NS record contains a typo in the domain name
  1. Domain name is not registrable

11 The .io Error – Taking Control of All .io Domains With a Targeted Registration https://thehackerblog.com/the-io-error-taking-control-of-all-io-domains-with-a-targeted-registration/

Zones and Delegations
Delegations Issues

- NS record points to an IP address
  - Not reachable and not valid
  - Reliabilty issue (e.g. other name server are not reachable/overloaded)

- NS record contains a typo in the domain name
  1. Domain name is not registrable
     - Reliabilty issue

11 The .io Error – Taking Control of All .io Domains With a Targeted Registration https://thehackerblog.com/the-io-error-taking-control-of-all-io-domains-with-a-targeted-registration/

Zones and Delegations
Delegations Issues

- NS record points to an IP address
  - Not reachable and not valid
  - Reliabilty issue (e.g. other name server are not reachable/overloaded)

- NS record contains a typo in the domain name
  1. Domain name is not registrable
     - Reliabilty issue
  2. Domain name is open registrable

     $\rightarrow$ Hijacking possibility[11]

11 The .io Error – Taking Control of All .io Domains With a Targeted Registration https://thehackerblog.com/the-io-error-taking-control-of-all-io-domains-with-a-targeted-registration/

- NS record points to an IP address
  - Not reachable and not valid
  - Reliabilty issue (e.g. other name server are not reachable/overloaded)

- NS record contains a typo in the domain name
  1. Domain name is not registrable
     - Reliabilty issue
  2. Domain name is open registrable

     $\rightarrow$ Hijacking possibility[11]
- NS record points to a host without a DNS service or without authoritative information on the zone (lame delegation)

11 The .io Error – Taking Control of All .io Domains With a Targeted Registration https://thehackerblog.com/the-io-error-taking-control-of-all-io-domains-with-a-targeted-registration/

- NS record points to an IP address
  - Not reachable and not valid
  - Reliabilty issue (e.g. other name server are not reachable/overloaded)

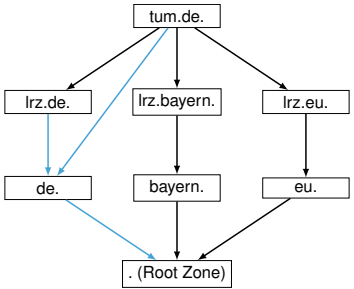
- NS record contains a typo in the domain name
  1. Domain name is not registrable
     - Reliabilty issue
  2. Domain name is open registrable

     $\rightarrow$ Hijacking possibility[11]
- NS record points to a host without a DNS service or without authoritative information on the zone (lame delegation)
  - E.g. `net.in.tum.de 3600 IN NS ns1.google.com`
  - `ns1.google.com` has no authoritative information on `net.in.tum.de`
  - Try `dig soa net.in.tum.de @ns1.google.com`

11 The .io Error – Taking Control of All .io Domains With a Targeted Registration https://thehackerblog.com/the-io-error-taking-control-of-all-io-domains-with-a-targeted-registration/

Trusted Computing Base (TCB)

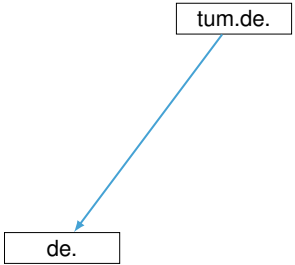
- A set of all components critical to a systems security
- First defined in the context of the kernel and trusted processes by John Rushby
- Ramasubramanian et al. defines[12]:
  The nameservers in the delegation graph of a domain name form the trusted computing base (TCB) of that name.
- More general: A zones TCB consists of all zones in the delegation graph



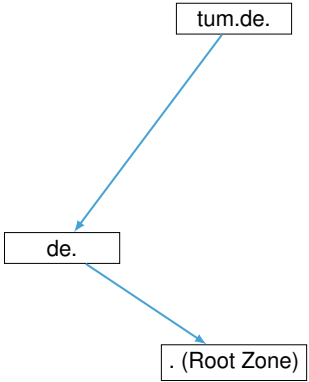
12 Ramasubramanian et al., Perils of Transitive Trust in the Domain Name Systemin ACM IMC 2005

Zones and Delegations
Example tum.de.

Zones and Delegations
Example tum.de.

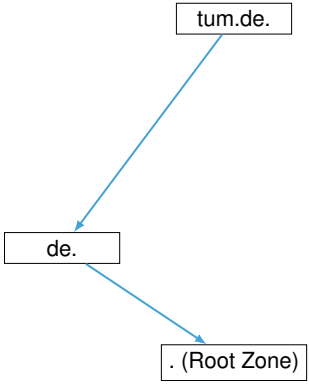


Zones and Delegations
Example tum.de.



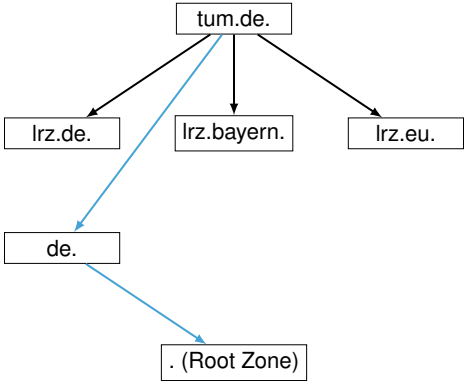
Zones and Delegations
Example tum.de.

tum.de.	86400	IN	NS	dns1.lrz.de.
tum.de.	86400	IN	NS	dns2.lrz.bayern.
tum.de.	86400	IN	NS	dns3.lrz.eu.



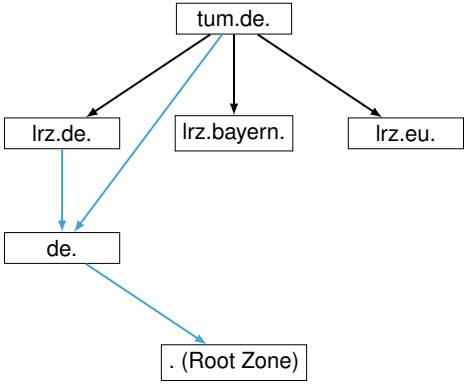
Zones and Delegations
Example tum.de.

tum.de.	86400	IN	NS	dns1.lrz.de.
tum.de.	86400	IN	NS	dns2.lrz.bayern.
tum.de.	86400	IN	NS	dns3.lrz.eu.



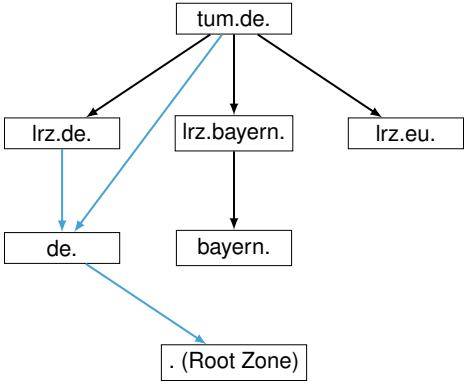
Zones and Delegations
Example tum.de.

tum.de.	86400	IN	NS	dns1.lrz.de.
tum.de.	86400	IN	NS	dns2.lrz.bayern.
tum.de.	86400	IN	NS	dns3.lrz.eu.



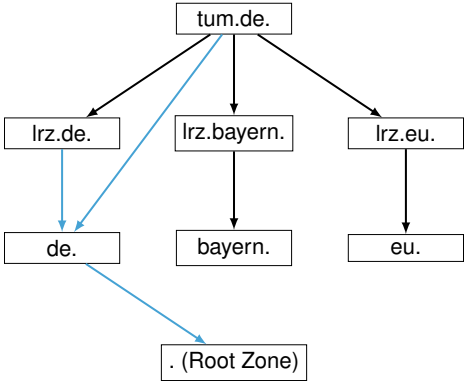
Zones and Delegations
Example tum.de.

tum.de.	86400	IN	NS	dns1.lrz.de.
tum.de.	86400	IN	NS	dns2.lrz.bayern.
tum.de.	86400	IN	NS	dns3.lrz.eu.



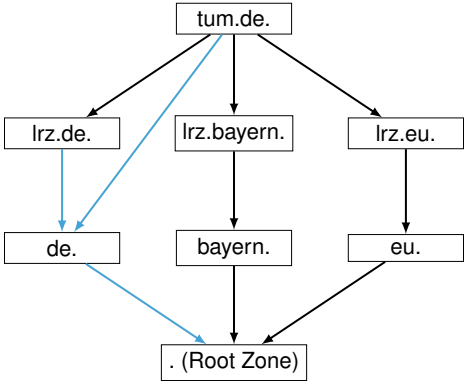
Zones and Delegations
Example tum.de.

tum.de.	86400	IN	NS	dns1.lrz.de.
tum.de.	86400	IN	NS	dns2.lrz.bayern.
tum.de.	86400	IN	NS	dns3.lrz.eu.



Zones and Delegations
Example tum.de.

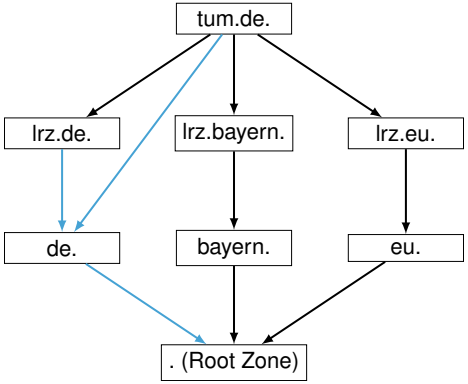
```
tum.de.   86400   IN   NS   dns1.lrz.de.
tum.de.   86400   IN   NS   dns2.lrz.bayern.
tum.de.   86400   IN   NS   dns3.lrz.eu.
```



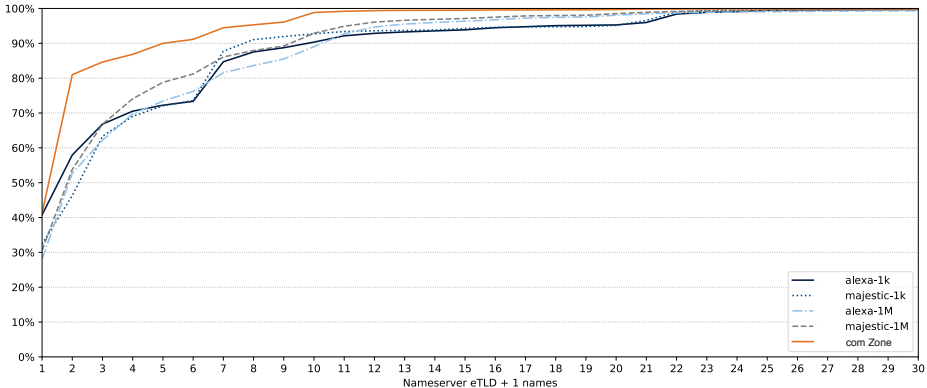
Zones and Delegations
Example tum.de.

tum.de.	86400	IN	NS	dns1.lrz.de.
tum.de.	86400	IN	NS	dns2.lrz.bayern.
tum.de.	86400	IN	NS	dns3.lrz.eu.

.bayern setup is more complex in reality

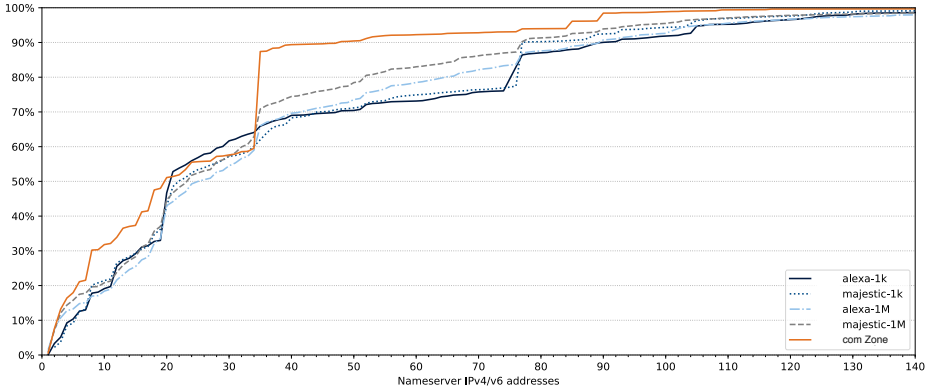


Zones and Delegations
TCB Statistics



- Idea: Number of eTLD + 1 label gives us an idea on the number of parties involved
- The more parties involved the higher is the attack surface
- Caveat: Some DNS provider use name server names in different eTLDs (e.g. AWS) $\rightarrow$ more eTLD + 1 names per provider
- Therefore: a lower number of eTLD + 1 names is better

Zones and Delegations
TCB Statistics



- Number of IP addresses per TCB is a more accurate representation for the number of hosts in the TCB (not considering anycast)
- Significant increases in the graph stem from DNS providers

Zones and Delegations
Name Server Resilience Requirements

RFC 8499 on zones:

> *Authoritative information is organized into units called ZONEs, and these zones can be automatically distributed to the name servers which provide redundant service for the data in a zone.*

Zones and Delegations
Name Server Resilience Requirements

RFC 8499 on zones:

> *Authoritative information is organized into units called ZONEs, and these zones can be automatically distributed to the name servers which provide redundant service for the data in a zone.*

Name servers which provide redundant service for the data in a zone.

RFC 8499 on zones:

> *Authoritative information is organized into units called ZONEs, and these zones can be automatically distributed to the name servers which provide redundant service for the data in a zone.*

Name servers which provide redundant service for the data in a zone.

RFC 2182 in 3.1:

> *Secondary servers must be placed at both topologically and geographically dispersed locations on the Internet, to minimise the likelihood of a single failure disabling all of them.*

Zones and Delegations
Name Server Resilience Requirements

RFC 8499 on zones:

> *Authoritative information is organized into units called ZONEs, and these zones can be automatically distributed to the name servers which provide redundant service for the data in a zone.*

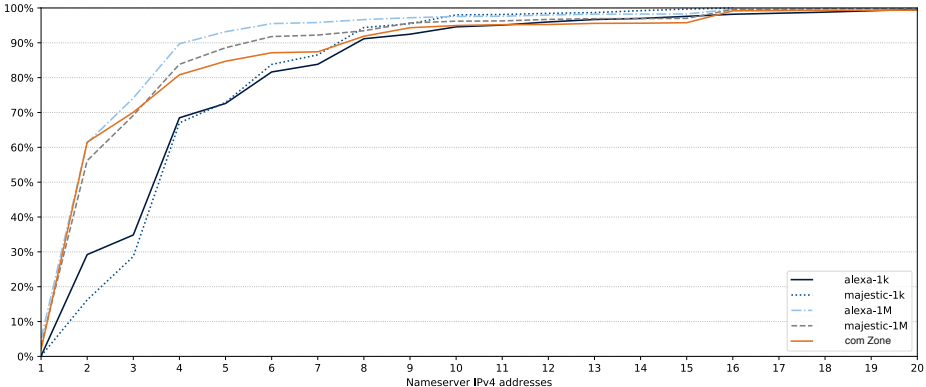
Name servers which provide redundant service for the data in a zone.

RFC 2182 in 3.1:

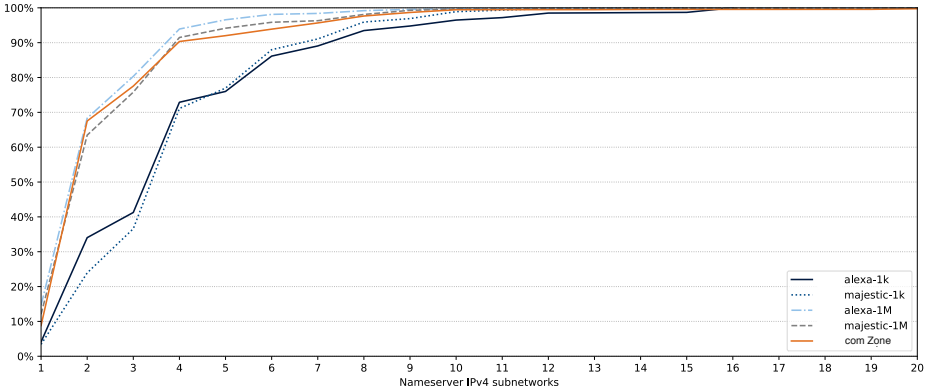
> *Secondary servers must be placed at both topologically and geographically dispersed locations on the Internet, to minimise the likelihood of a single failure disabling all of them.*

Servers must be placed at both topologically and geographically dispersed

Nameserver IPv4 addresses per zone



Nameserver IPv4 /24 subnets per zone



- Most zones have two nameserver IP addresses
- Aggregated on /24 subnets for topological diversity we can find up to 10 % of non compliant zones

Domain Name System

EDNS
Transport Protocol

- Default is UDP
- DNS UDP supports messages up to 512 byte payload
- With additions such as DNSSEC and EDNS0 the boundary of 512 bytes is easily broken
- UDP-Fragmentation does not work reliable
- When it works it can be abused [1]

Fallback to TCP

- DNS standard included TCP from the beginning (optional)
- DNS Flag Day 2020 tries to force all DNS infrastructure provider to support TCP [2]
- TCP needs an extra RTT to setup connection

[1] A. Herzberg and H. Shulman, Fragmentation Considered Poisonous, or: One-domain-to-rule-them-all.org, 2013 IEEE CNS
[2] DNS Flag Day 2020, https://dnsflagday.net/2020/

EDNS

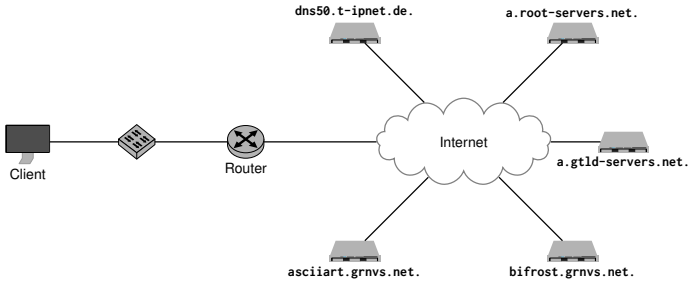
Extension mechanisms for DNS (EDNS(0))

- Defined in RFC6891
- Backwards compatible (Fallback mechanism required)
- Advertises size of maximum UDP payload size
- Extend 4 bit RCODE
- Adds new label types
- Adds the OPT pseudo-RR

Extension mechanisms for DNS (EDNS(0))

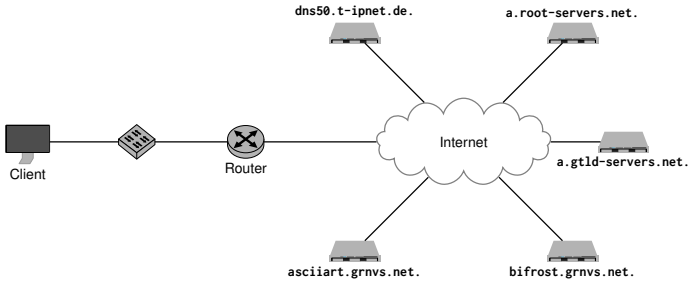
- Defined in RFC6891
- Backwards compatible (Fallback mechanism required)
- Advertises size of maximum UDP payload size
- Extend 4 bit RCODE
- Adds new label types
- Adds the OPT pseudo-RR
  - RR in the *Additional* section (maximum one is allowed)
  - Always related to the message it is in
  - Shall never be cached
  - TTL is partly used for extenden RCODE
  - RDATA contains key-value pairs

EDNS
EDNS Client Subnet (ECS)



- Defined in RFC7871 with EDNS OPTION-CODE 8
- Resolver forwards the client IP address to the authoritative name server
- Sends:

EDNS
EDNS Client Subnet (ECS)



- Defined in RFC7871 with EDNS OPTION-CODE 8
- Resolver forwards the client IP address to the authoritative name server
- Sends:
  - IP address family
  - Source prefix length (number of relevant bits in the IP address)
  - Scope prefix length (number of bits the response covers)
  - IP address

EDNS
ECS Advanced

- Recursive resolvers can forward ECS requests

  Usefull for architectures including forwarder

- Recursive resolvers can forward ECS requests

  Usefull for architectures including forwarder

- Caching policy:
  - Source prefix length denotes the maximum cachable

- Recursive resolvers can forward ECS requests

  Usefull for architectures including forwarder

- Caching policy:
  - Source prefix length denotes the maximum cachable
  - Source prefix length > Scope prefix length
    - Less bits are needed for the best respone
    - Cache answer for address with scope prefix length

- Recursive resolvers can forward ECS requests

  Usefull for architectures including forwarder

- Caching policy:
  - Source prefix length denotes the maximum cachable
  - Source prefix length > Scope prefix length
    - Less bits are needed for the best respone
    - Cache answer for address with scope prefix length
  - Source prefix length < Scope prefix length
    - Source prefix length was not specific enough to select the most appropriate response
    - Resolver can retry query with longer prefix $\rightarrow$ better user experience
    - Or cache the answers for request matching the exact prefix and source prefix length

DNS Security
Problem

Original design of DNS does not include any security features

- Focus on scalability and distribution
- DNS does not provide a mechanism to authenticate replies
- The integrity of replies is not protected
- Client privacy is not given
  - Queries are sent in plain text
  - Queries reveal information about client behavior/traffic

DNS Security Solutions

Protocols have been developed to solve different security issues:

- DNSSEC
  - Provides authenticity and integrity of DNS responses
- DNS Encryption
  - Protects the privacy of a client
  - Encrypts the traffic between client and resolver
  - E.g., DNS over TLS (DoT), DNS over HTTPS (DoH)
- QNAME Minimization
  - Protects the privacy of a client
  - Reduces the information sent to name servers

Domain Name System Security Extensions (DNSSEC)

- Sign DNS records
  - Public-key cryptography
  - Verified public keys of the DNS root zone (Trusted Third Party)
  - Authentication chain of trust from root zone to child zone
- Additional DNS RRs to integrate DNSSEC, e.g.,
  - RRSIG (Resource Record Signature)
  - DNSKEY (Public Key)
  - NSEC/NSEC3 (Next secure record (v3))

Table 1: DNSSEC amongst the top 20 DNS operators.

Operator	Domains	Unsigned		Secured		Invalid		Islands	
		Num	%	Num	%	Num	%	Num	%
GoDaddy	56 446 359	56 326 752	99.8	107 550	0.2	8 550	0.02	3 507	0.01
Cloudflare	27 790 208	26 541 985	95.5	799 377	2.9	16 694	0.1	432 152	1.6
Namecheap	10 252 586	10 119 070	98.7	126 601	1.2	5 300	0.05	1 615	0.02
Google Domain (SquareSpace)	9 931 131	5 197 647	52.3	4 496 848	45.3	109 499	1.1	127 137	1.3
WIX	7 318 524	5 989 947	81.8	74 423	2.3	2 954	0.04	1 151 200	15.7
Hostinger	6 561 661	6 556 301	99.9			5 360	0.1		
AfterNIC (GoDaddy)	5 360 163	5 349 129	99.8			11 034	0.2		
HiChina	4 637 997	4 628 516	99.8			9 481	0.2		
AWS	3 698 499	3 653 373	98.8	30 005	0.8	4 345	0.1	10 776	0.3
GName	3 558 801	3 556 082	99.9	1 145	0.03	1 002	0.03	572	0.02
NameBright	3 516 303	3 515 548	99.98	73	0.002	680	0.02	2	0.000 1
SquareSpace	2 735 515	2 710 040	99.1	24 278	0.9	1 023	0.04	174	0.01
OVH	2 662 864	1 469 425	55.2	1 169 714	43.9	2 839	0.1	20 886	0.8
Sedo	2 340 028	2 336 383	99.8			3 645	0.2		
BlueHost	1 976 091	1 960 552	99.2	13 188	0.7	136	0.06	1 215	0.06
NameSilo	1 847 474	1 846 251	99.9			1 223	0.1		
Alibaba	1 570 903	1 564 980	99.6	2 675	0.2	1 216	0.1	2 032	0.13
DynaDot	1 552 892	1 552 431	99.97			461	0.03		
Wordpress	1 549 730	1 541 499	99.5	7 824	0.5	347	0.02	60	0.004
SiteGround	1 535 176	1 533 874	99.92			1 302	0.08		

- 93.2 % (628.1 M) zones do not use DNSSEC
- 5.5 % (15.8 M) zones were correctly singed use DNSSEC
- 0.2 % (640 k) zones fail validation

⟶ DNSSEC deployment remains rare

What's next in making Encrypted DNS-over-HTTPS the Default

Selena Deckelmann | **September 6, 2019**

In 2017, Mozilla began working on the DNS-over-HTTPS (DoH) protocol, and since June 2018 we've been running experiments in Firefox to ensure the performance and user experience are great. We've also been surprised and excited by the more than 70,000 users who have already chosen on their own to explicitly enable DoH in Firefox Release edition. We are close to releasing DoH in the USA, and we have a few updates to share.

13 https://blog.mozilla.org/futurereleases/2019/09/06/whats-next-in-making-dns-over-https-the-default/
14 https://www.zdnet.com/article/uk-isp-group-names-mozilla-internet-villain-for-supporting-dns-over-https/
15 https://www.zdnet.com/article/dns-over-https-google-hits-back-at-misinformation-and-confusion-over-its-plans/

What's next in making Encrypted DNS-over-HTTPS the Default

Selena Deckelmann | **September 6, 2019**

In 2017, Mozilla began working on the DNS-over-HTTPS (DoH) protocol, and since June 2018 we've been running experiments in Firefox to ensure the performance and user experience are great. We've also been surprised and excited by the more than 70,000 users who have already chosen on their
close to releasin

UK ISP group names Mozilla 'Internet Villain' for supporting 'DNS-over-HTTPS'

UK government and local ISPs are putting the pressure on browsers to drop plans to support DoH protocol.

13 https://blog.mozilla.org/futurereleases/2019/09/06/whats-next-in-making-dns-over-https-the-default/
14 https://www.zdnet.com/article/uk-isp-group-names-mozilla-internet-villain-for-supporting-dns-over-https/
15 https://www.zdnet.com/article/dns-over-https-google-hits-back-at-misinformation-and-confusion-over-its-plans/

What's next in making Encrypted DNS-over-HTTPS the Default

Selena Deckelmann | **September 6, 2019**

In 2017, Mozilla began working on the DNS-over-HTTPS (DoH) protocol, and since June 2018 we've been running experiments in Firefox to ensure the performance and user experience are great. We've also been surprised and excited by the more than 70,000 users who have already chosen on their
close to releasin

UK ISP group names Mozilla 'Internet Vi

DNS over HTTPS: Google hits back at 's' UK g 'misinformation and confusion' over its ocol. plans

Google sets the record straight on its plans to encrypt DNS requests from Chrome over HTTPS.

13 https://blog.mozilla.org/futurereleases/2019/09/06/whats-next-in-making-dns-over-https-the-default/
14 https://www.zdnet.com/article/uk-isp-group-names-mozilla-internet-villain-for-supporting-dns-over-https/
15 https://www.zdnet.com/article/dns-over-https-google-hits-back-at-misinformation-and-confusion-over-its-plans/

DNS Encryption resulted in a heated discussion in the media:

- What are possible solutions?
- Which properties do they promise?
- What are the advantages and disadvantages?
- What is **not** solved by these solutions?

Problem Statement:

- Queries in plain text reveal user behavior and accessed services
- Nearly everything in the Internet relies on DNS
- Intercepting client traffic enables detailed fingerprinting

Goals:

- DNS encryption only targets the communication between client and resolver
- Recursive queries from resolver to name servers are still plain text
- These queries should not contain client information
- DNS resolution itself is not altered

Assumptions:

- Resolvers can be trusted
- Resolvers are used by a large number of clients

DNSCrypt[16]

- Development started in 2008
- Own protocol for encryption and authentication
- Supports UDP and TCP with port 443

DNS-over-TLS [7]

- Uses existing protocol TLS for encryption
- Based on TCP instead of UDP
- Uses port 853 (Critics: can be blocked)

16 https://dnscrypt.info/

DNS Security
DNS Encryption: DNS over HTTPS [8]

- Uses HTTPS for communication and encryption
- Based on TCP instead of UDP
- Uses port 443 (hard to block)
- Can be configured individually by applications in user space
- DoH service can be hosted on theoretically arbitrary path: e.g. URI-template: `https://dnsserver.example.net/dns-query{?dns}`
- Can be performed using two methods:

  HTTP `GET` Method Path contains the DNS query base64url-encoded
  HTTP `POST` Method DNS message in send as the post content

DNS Security
DNS Encryption: DNS over HTTPS: Examples [8]

`GET` Query

```
:method = GET
:scheme = https
:authority = dnsserver.example.net
:path = /dns-query?dns=AAABAAABAAAAAAAAA3d3dwdleGFtcGxlA2NvbQAAAQAB
accept = application/dns-message
```

`POST` Query

```
:method = POST
:scheme = https
:authority = dnsserver.example.net
:path = /dns-query
accept = application/dns-message
content-type = application/dns-message
content-length = 33

<33 bytes represented by the following hex encoding>
00 00 01 00 00 01 00 00  00 00 00 00 03 77 77 77
07 65 78 61 6d 70 6c 65  03 63 6f 6d 00 00 01 00
01
```

Answer

```
:status = 200
content-type = application/dns-message
content-length = 61
cache-control = max-age=3709

<61 bytes represented by the following hex encoding>
00 00 81 80 00 01 00 01  00 00 00 00 03 77 77 77
07 65 78 61 6d 70 6c 65  03 63 6f 6d 00 00 1c 00
01 c0 0c 00 1c 00 01 00  00 0e 7d 00 10 20 01 0d
b8 ab cd 00 12 00 01 00  02 00 03 00 04
```

DNS Security
DNS Encryption

Pros:

- Client traffic is encrypted

Cons:

- Internal DNS configurations might be overwritten

Debatable:

- DoH/DoT is faster?
  - TLS/HTTPS is fast and well studied but DNS (UDP/53) as well
- DoH/DoT prevents censorship?
  - The behavior of a resolver is unchanged
  - Probably more clients use large, international resolvers in the future
  - **But** they can censor as well or might be forced to by governments
- DoH/DoT prevents collection of your data?
  - Data can still be collected by the resolver

Encrypting DNS traffic between a client and resolver improves the privacy of clients by preventing the effectiveness of eavesdropping traffic, but:

- You still have to trust the resolver
- Data can still be collected
- Censorship is still possible
- Only eavesdropping traffic is limited

DNS Security
DNS Encryption: Discovery of Designated Resolvers[17]

- Most network provisioning protocols like DHCP or IPv6 RA only provide the resolvers IP
- DNS encryption mechanisms like DoT, DoH or DoQ may need more configuration information:
  - Resolver Hostname
  - non-default port numbers
  - URI-templates (e.g. DoH query-path)
  - Alternative IP addresses

17 RFC 9462 https://www.rfc-editor.org/rfc/rfc9462.html

DNS Security
DNS Encryption: Discovery of Designated Resolvers[17]

- Most network provisioning protocols like DHCP or IPv6 RA only provide the resolvers IP
- DNS encryption mechanisms like DoT, DoH or DoQ may need more configuration information:
  - Resolver Hostname
  - non-default port numbers
  - URI-templates (e.g. DoH query-path)
  - Alternative IP addresses
- Idea: let the client discover information on designated encrypted DNS resolvers by querying for `SVCB` records
- Clients should query the resolver for DNS `SVCB` records when configured with an unencrypted DNS resolver IP address using the special-use domain name (SUDN) `resolver.arpa`.
- Two types: Verified Discovery and Opportunistic Discovery

```
; <<>> DiG 9.18.41 <<>> _dns.resolver.arpa. SVCB @8.8.8.8 +noadditional +nostats +noquestion
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 28527
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 4

;; ANSWER SECTION:
_dns.resolver.arpa.     86400   IN      SVCB    1 dns.google. alpn="dot"
_dns.resolver.arpa.     86400   IN      SVCB    2 dns.google. alpn="h2,h3" key7="/dns-query{?dns}"
```

[17] RFC 9462 https://www.rfc-editor.org/rfc/rfc9462.html

DNS Security
QNAME Minimization

Problem:

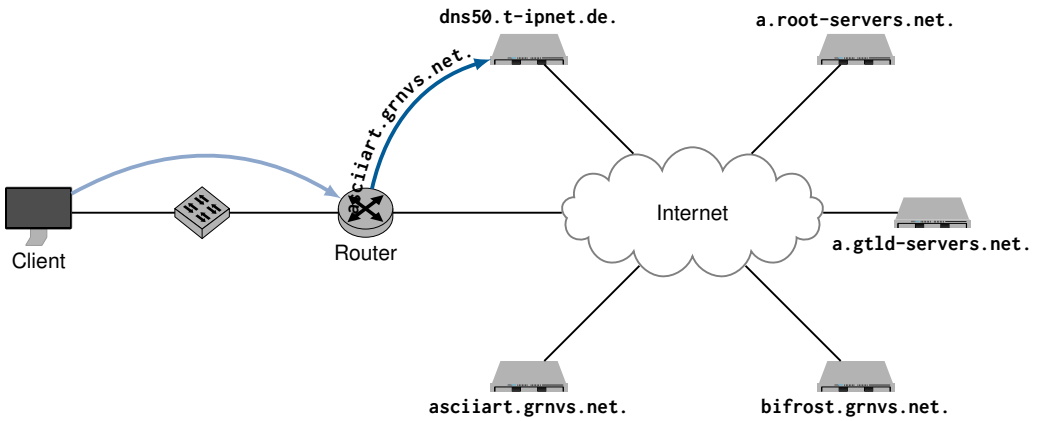
- Resolvers initially sent the complete QNAME and requested QTYPE to all name servers
- Each name server during the recursive resolution learns about the QNAME and QTYPE

Solution:

- DNS Query Name Minimisation RFC7816 [9]
- Send the exact QNAME and QTYPE only to the authoritative NS
- Only resolve the authoritative NS for each label during the recursive resolution

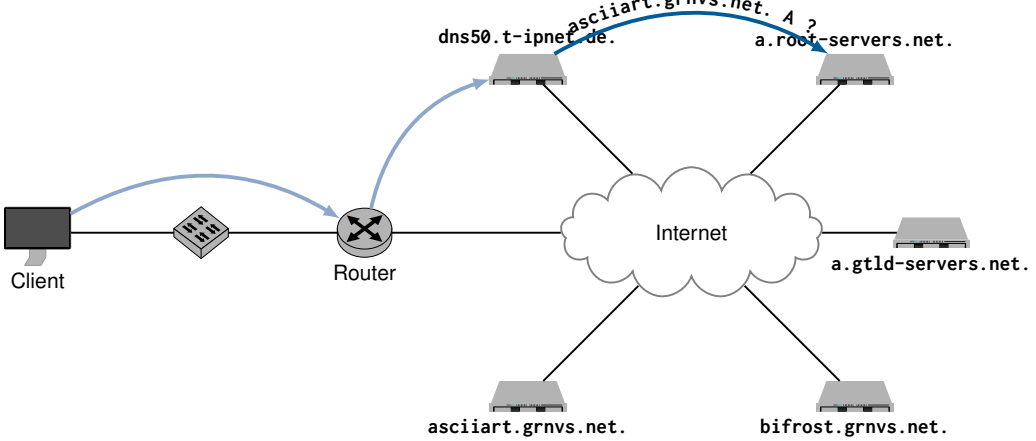
DNS Security
QNAME Minimization

Example:



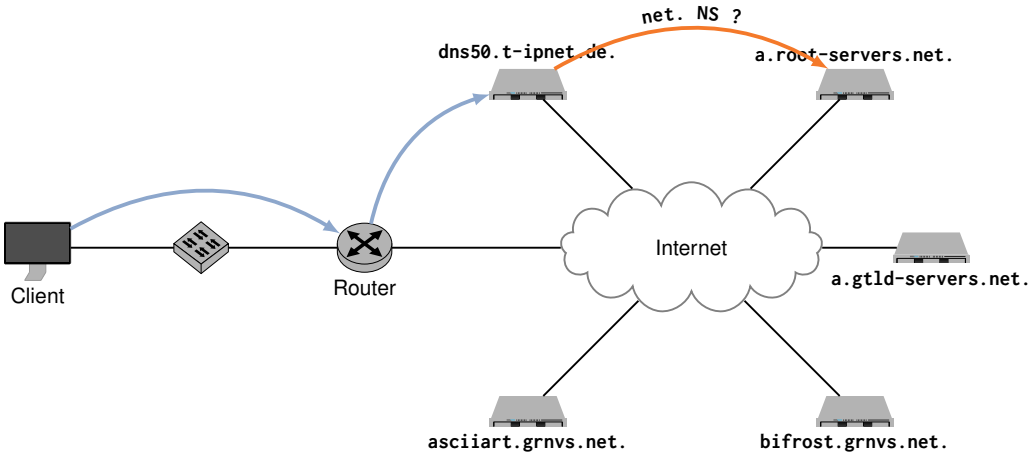
DNS Security
QNAME Minimization

Example:



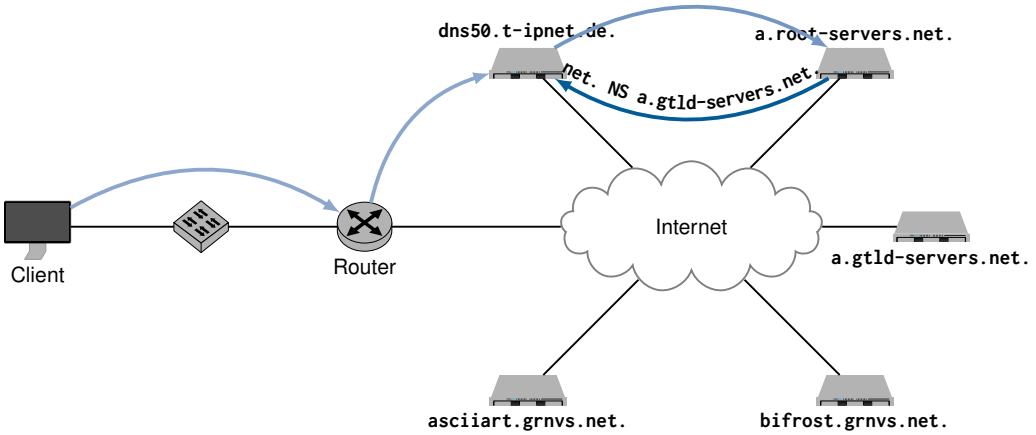
DNS Security
QNAME Minimization

Example:



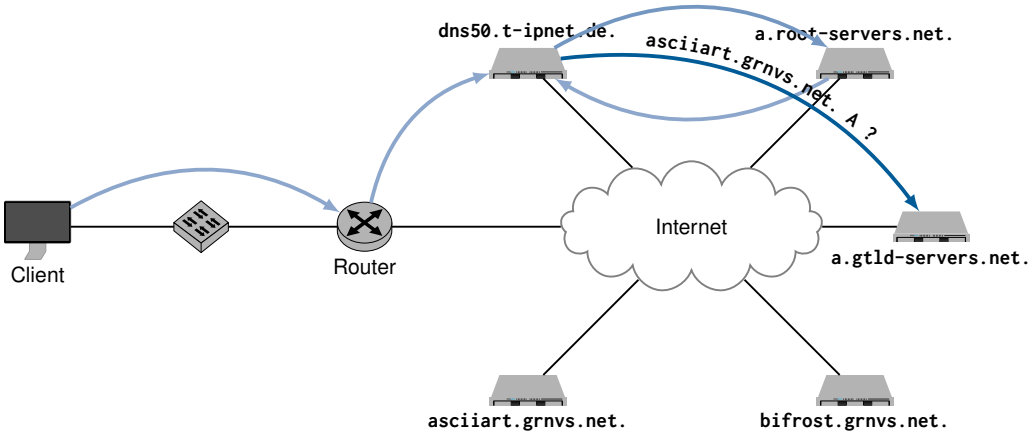
DNS Security
QNAME Minimization

Example:



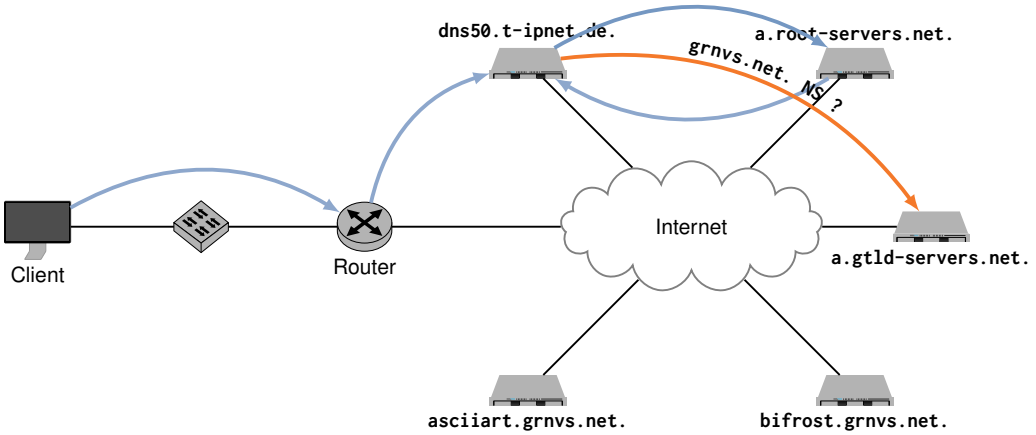
DNS Security
QNAME Minimization

Example:



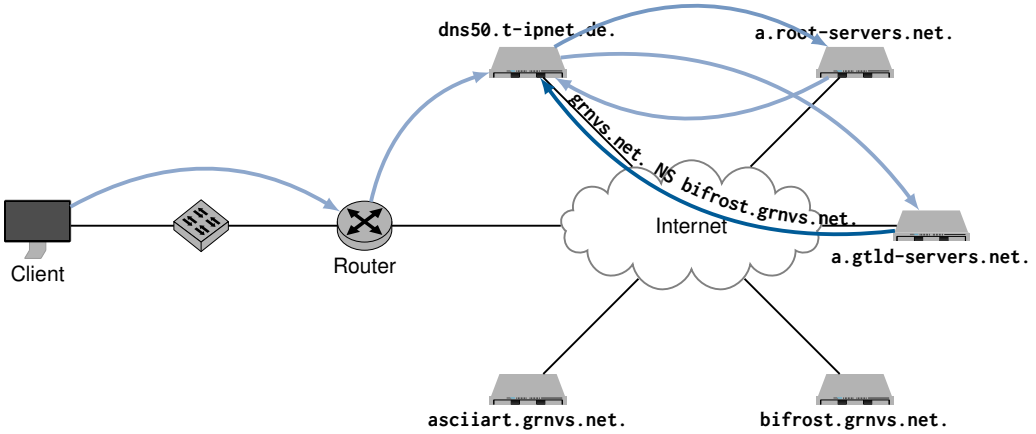
DNS Security
QNAME Minimization

Example:



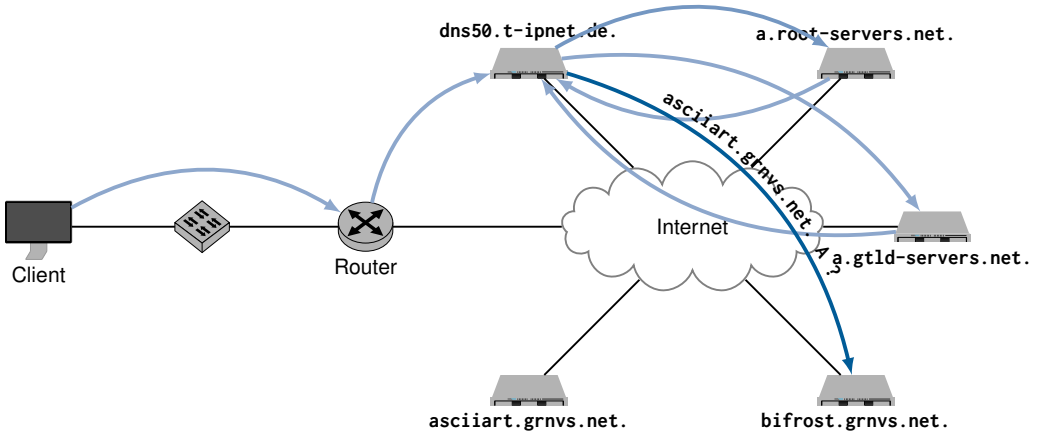
DNS Security
QNAME Minimization

Example:



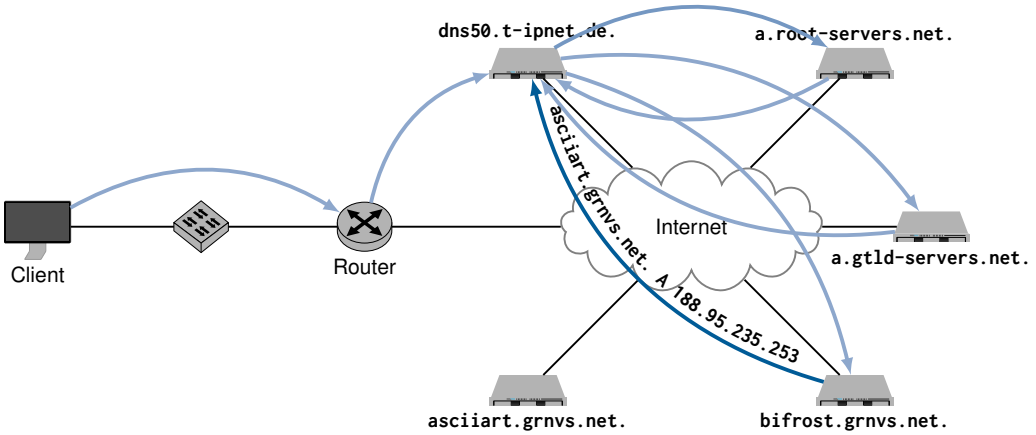
DNS Security
QNAME Minimization

Example:



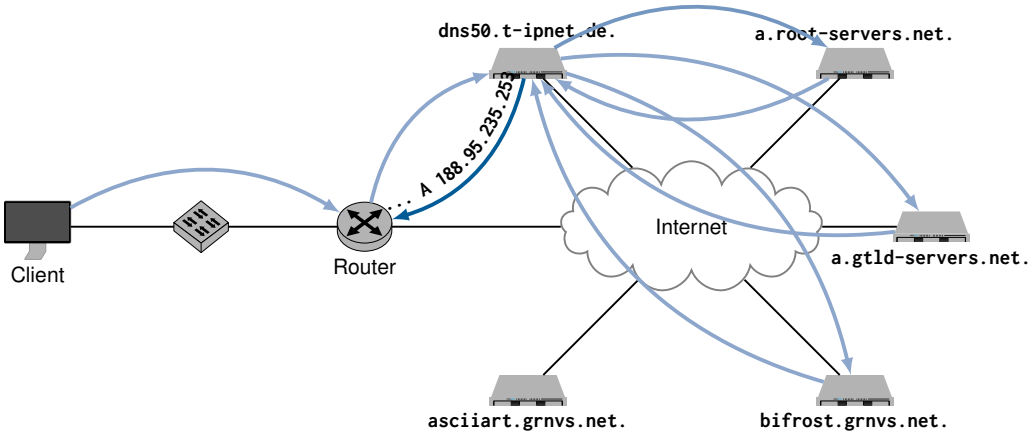
DNS Security
QNAME Minimization

Example:



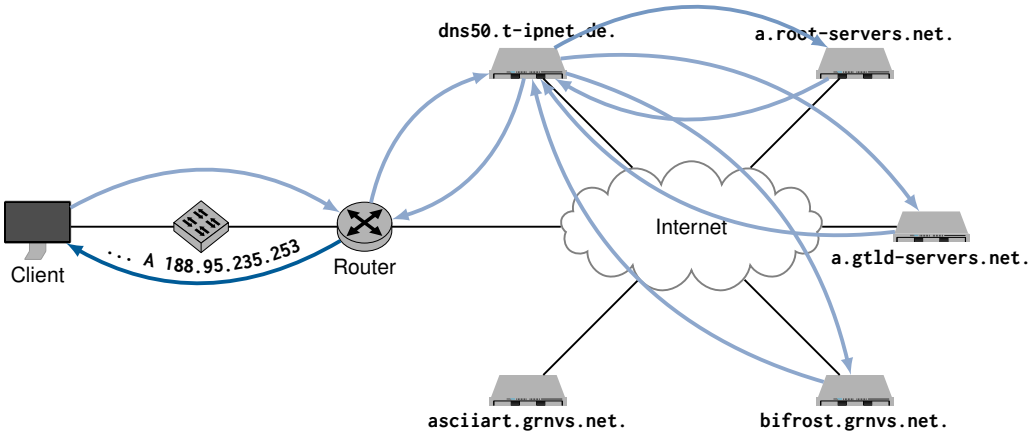
DNS Security
QNAME Minimization

Example:



DNS Security
QNAME Minimization

Example:



DNS Security
QNAME Minimization

QNAME Minimization only changes the resolver behavior and basically follows the DNS specification, but:

- Increased rate of unsuccessful queries (up to 5%[10])
  - Some NS incorrectly reply to NS queries (REFUSED)
  - → Use different QTYPE (A, AAAA)
  - Some NS incorrectly reply to emtpy labels (no data for name)
  - → Fallback to query with all labels
- Increased query load (up to 26% [10])
  - All labels have to be queried one by one
  - A NS authoritative for multiple labels could reply with most significant reply if full name is known
  - → Fallback to query with all labels when same NS is queried

→ Deployment of QNAME minimization is hindered by NS miss-configurations

→ Resolver implement algorithms with different fallback behavior

Domain Name System

[1] P. E. Hoffman and K. Fujiwara, DNS Terminology, RFC 9499, Mar. 2024. DOI: 10.17487/RFC9499. [Online]. Available: https://www.rfc-editor.org/info/rfc9499.

[2] P. Moackapetris, Domain Names – Concepts and Facilities, https://tools.ietf.org/html/rfc1034, 1987.

[3] P. Moackapetris, Domain Names – Implementation and Specification, https://tools.ietf.org/html/rfc1035, 1987.

[4] T. Lauinger, K. Onarlioglu, A. Chaabane, W. Robertson, and E. Kirda, "WHOIS Lost in Translation: (Mis)Understanding Domain Name Expiration and Re-Registration," in Proceedings of the 2016 Internet Measurement Conference, ser. IMC '16, Santa Monica, California, USA: Association for Computing Machinery, 2016, 247–253, ISBN: 9781450345262. DOI: 10.1145/2987443.2987463. [Online]. Available: https://doi.org/10.1145/2987443.2987463 (visited on 03/27/2024).

[5] Internet Corporation for Assigned Names and Numbers (ICANN), Expired Registration Recovery Policy, Feb. 2013. [Online]. Available: https://www.icann.org/resources/pages/errp-2013-02-28-en (visited on 02/26/2024).

[6] Q Misell, F. Steurer, J. Zirngibl, A. Feldmann, and T. Fiebig, "Measuring the deployment of dnssec bootstrapping using authenticated signals," in Proceedings of the 2025 ACM Internet Measurement Conference, ser. IMC '25, USA: Association for Computing Machinery, 2025, 1002–1009, ISBN: 9798400718601. DOI: 10.1145/3730567.3764501. [Online]. Available: https://doi.org/10.1145/3730567.3764501.

[7] Z. Hu, L. Zhu, J. Heidemann, A. Mankin, D. Wessels, and P. Hoffman, "Specification for dns over transport layer security (tls)," RFC 7858, 2016.

[8] P. Hoffman and P. McManus, "Dns queries over https (doh)," RFC 8484, 2018.

[9] S. Bortzmeyer, “DNS Query Name Minimisation to Improve Privacy,” RFC 7816, 2016. [Online]. Available: https://rfc-editor.org/rfc/rfc7816.txt.

[10] W. B. de Vries, Q. Scheitle, M. Müller, W. Toorop, R. Dolmans, and R. van Rijswijk-Deij, “A first look at qname minimization in the domain name system,” in Passive and Active Measurement, D. Choffnes and M. Barcellos, Eds., Cham: Springer International Publishing, 2019, pp. 147–160, ISBN: 978-3-030-15986-3.