DNS

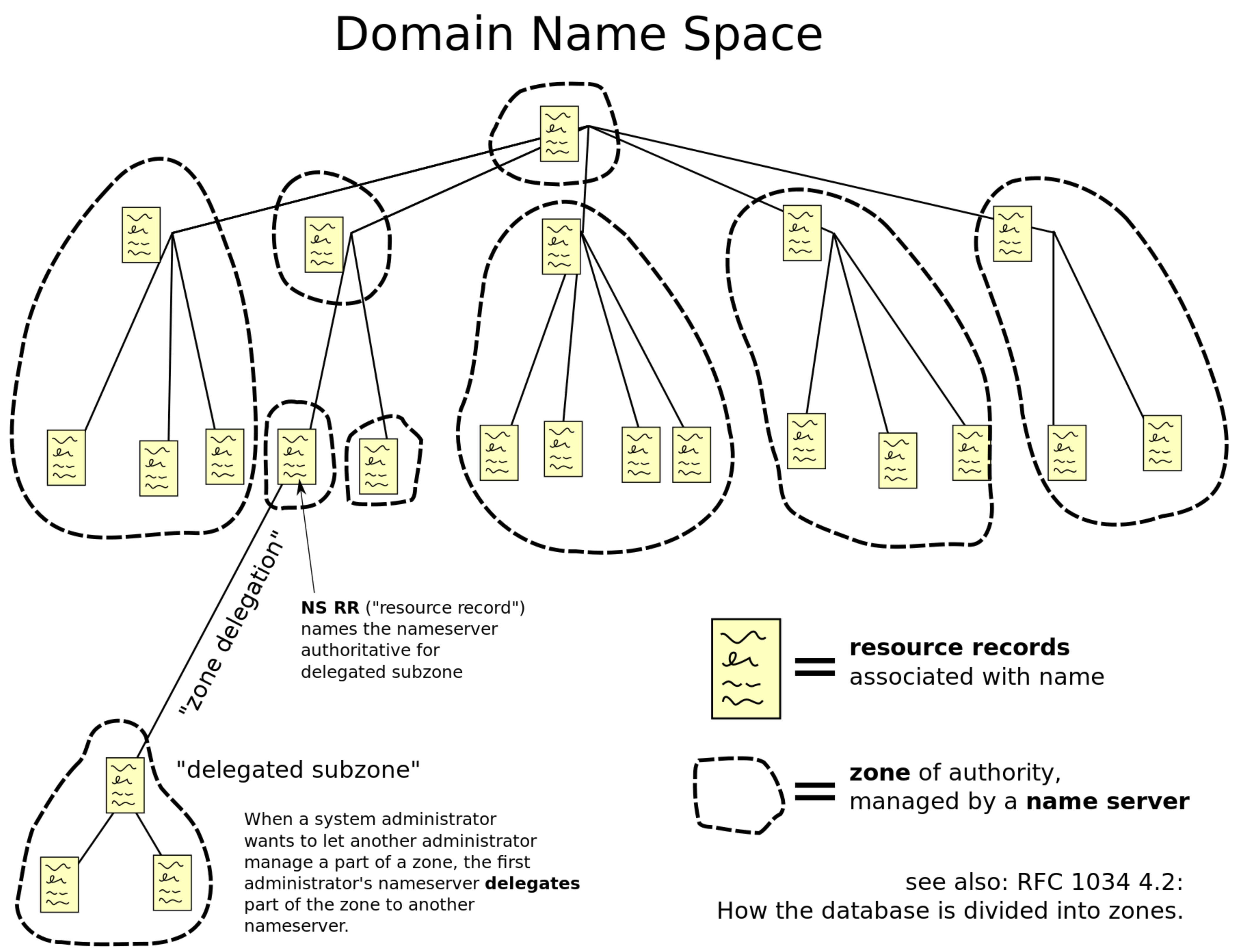
Network Security

What problem is solved by DNS?

- On the network layer, devices are usually identified by their IP addresses:
    - 147.188.128.127
    - 2001:db8:85a3:8d3:1319:8a2e:370:7348
- These are very inconvenient and can change sometimes!
- DNS allows to give out names (domains) that can point to IP addresses.
- These names are also used for various other purposes (such as emails).

DNS

- DNS is a hierarchical, decentralised naming system, i.e., a decentralised database storing names and associated data.
- DNS consists of name servers (NS).
- Each domain is assigned an **authoritative name server**, and can delegate parts of their responsibilities to other name servers.


Domain Name Space
"zone delegation"
"delegated subzone"
= **resource records** associated with name
= **zone** of authority, managed by a **name server**

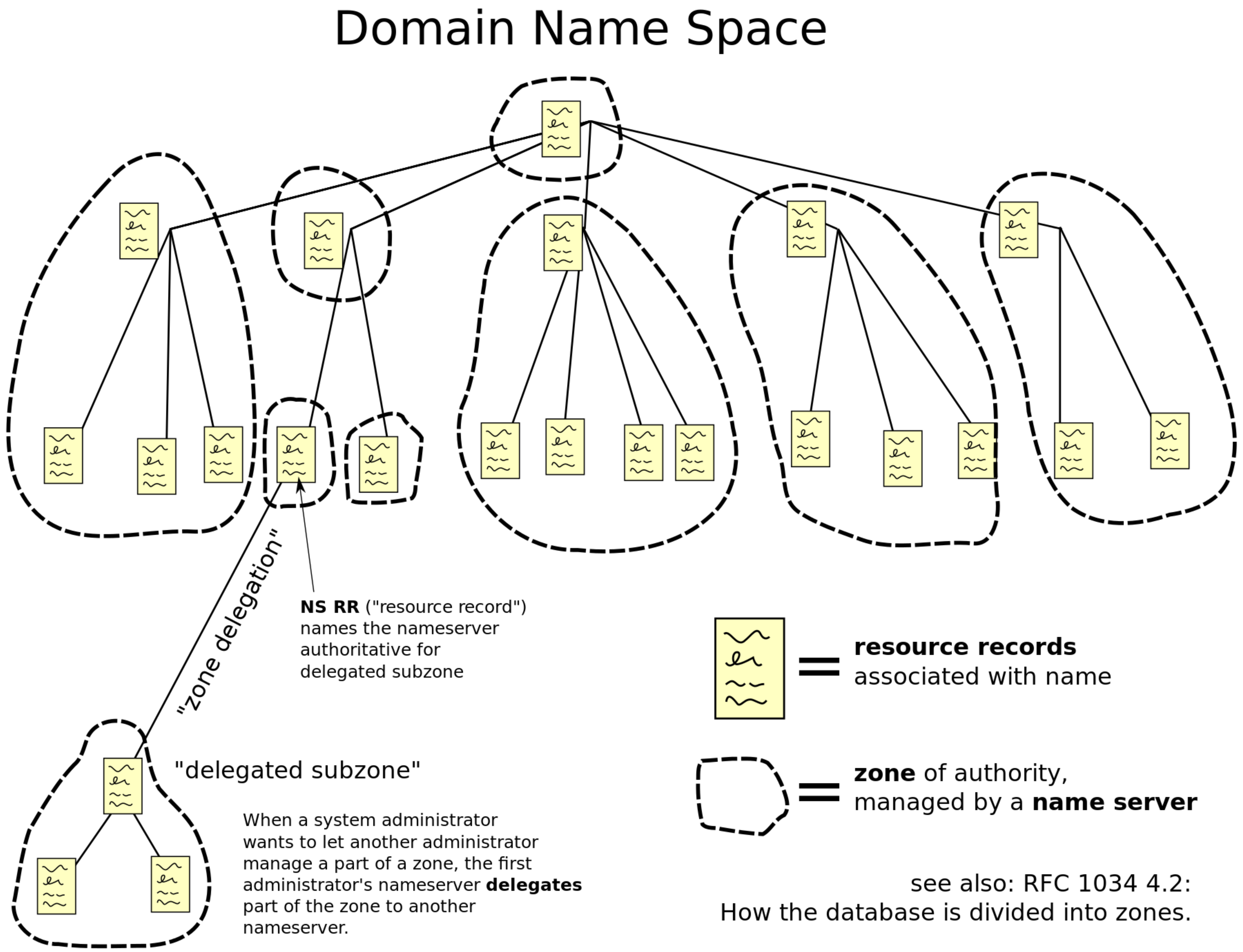
NS RR ("resource record") names the nameserver authoritative for delegated subzone

When a system administrator wants to let another administrator manage a part of a zone, the first administrator's nameserver **delegates** part of the zone to another nameserver.

see also: RFC 1034 4.2:
How the database is divided into zones.

DNS

- The database holds resource records containing information about the domains. Several types of resource records exist, e.g.,
  - *A*/*AAAA* to map domains to IP addresses
  - *MX* to identify the systems handling email for this domain
  - *TXT* to add arbitrary comments (sometimes used for other systems).

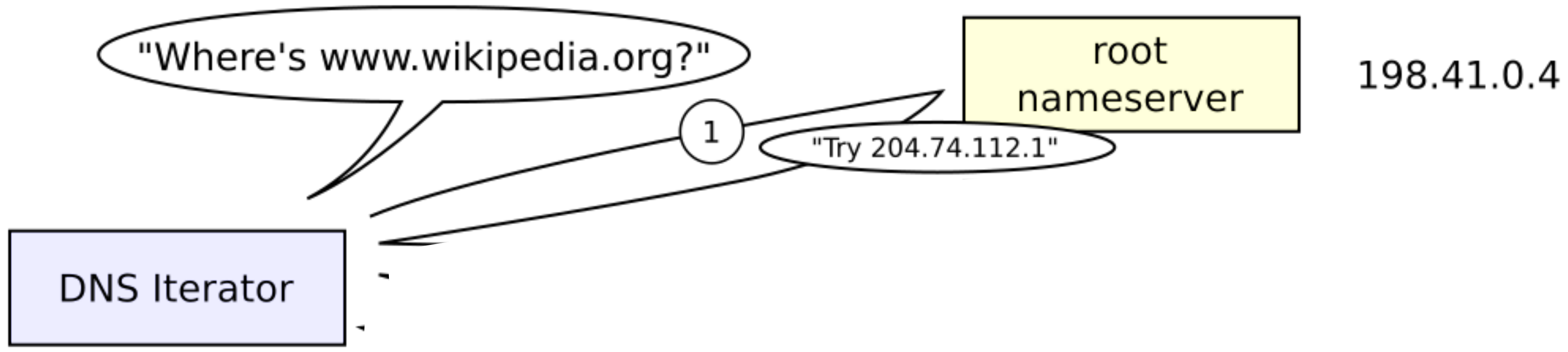

Domain Name Space
"zone delegation"
NS RR ("resource record") names the nameserver authoritative for delegated subzone
"delegated subzone"
= resource records associated with name
= zone of authority, managed by a name server

When a system administrator wants to let another administrator manage a part of a zone, the first administrator's nameserver **delegates** part of the zone to another nameserver.

see also: RFC 1034 4.2: How the database is divided into zones.

DNS Example

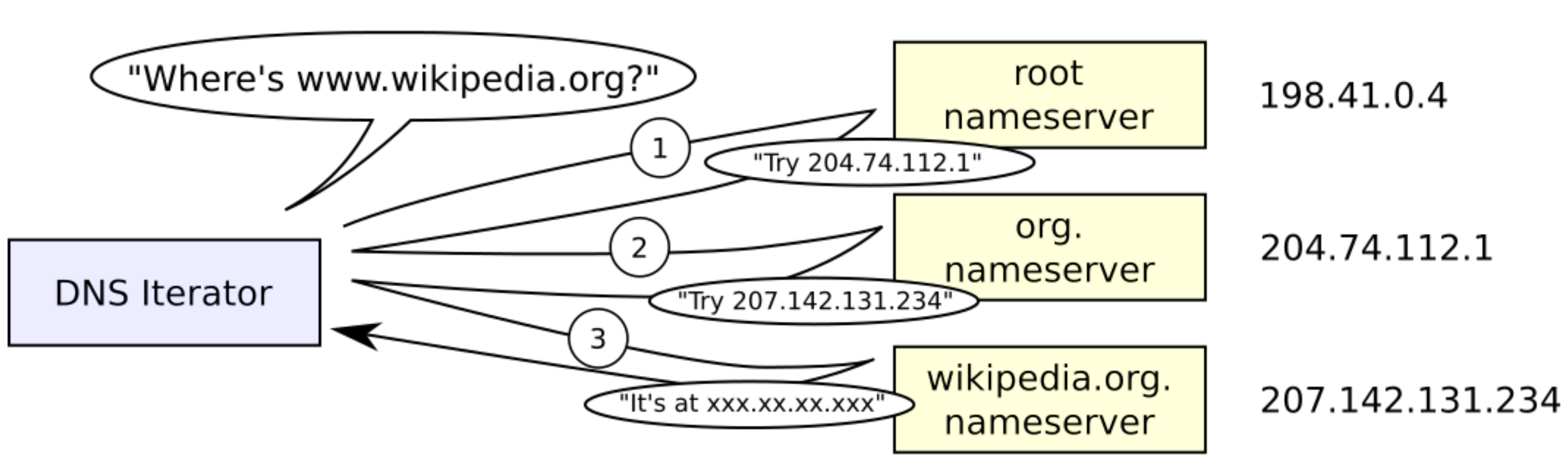
- Example: the root NS delegates the *org* top-level domain to a specific *org* NS.
- The *org* NS delegates the *wikipedia.org* domain to another NS.
- The *wikipedia.org* NS is the **authoritative NS** for this domain.
  - In this example, it answers the query immediately.
  - It could also delegate some subdomains to other name servers.



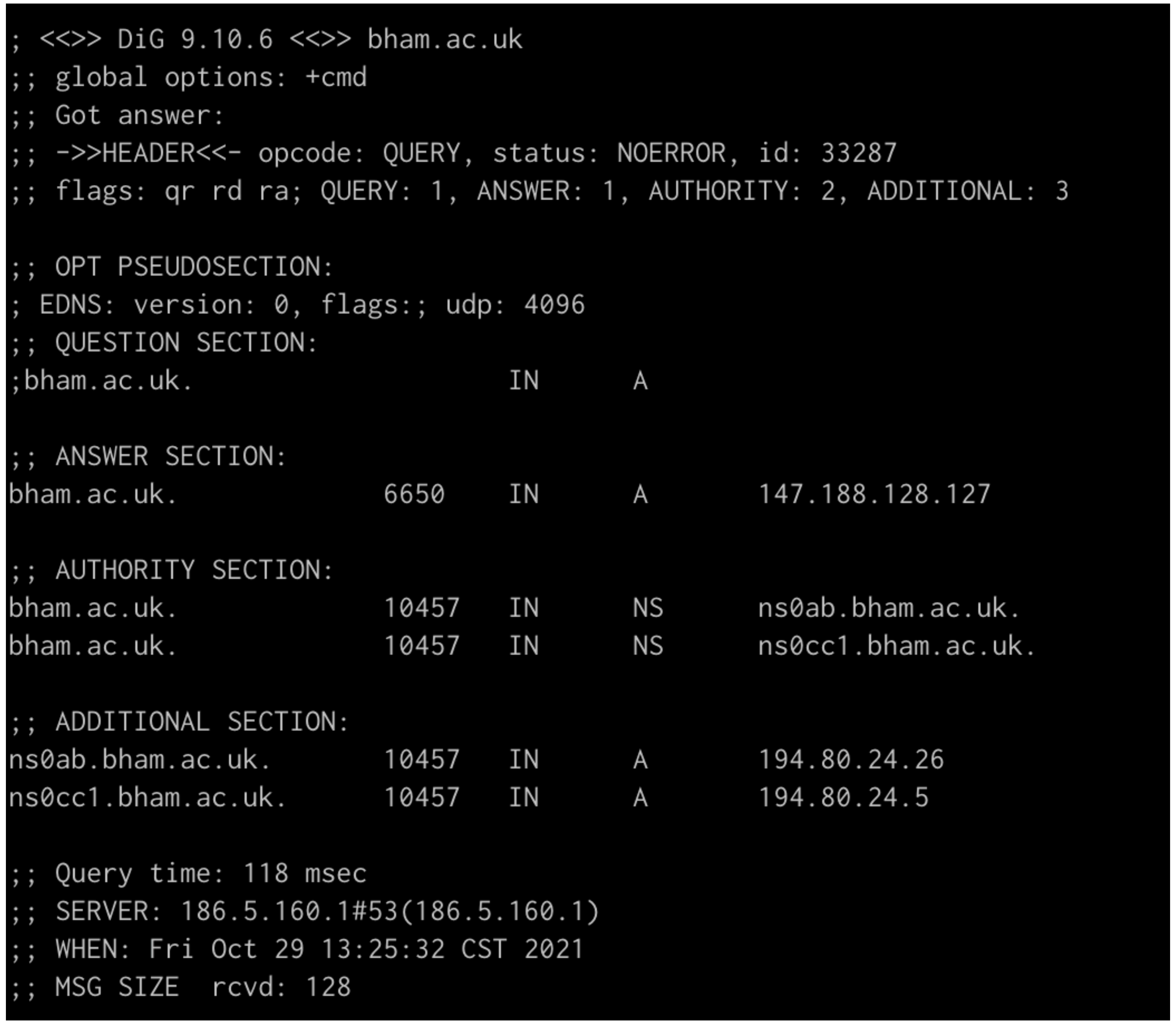
DNS Caching

- To avoid running such a recursive lookup for every domain, DNS employs **caching name servers** and devices usually have **local caches**.

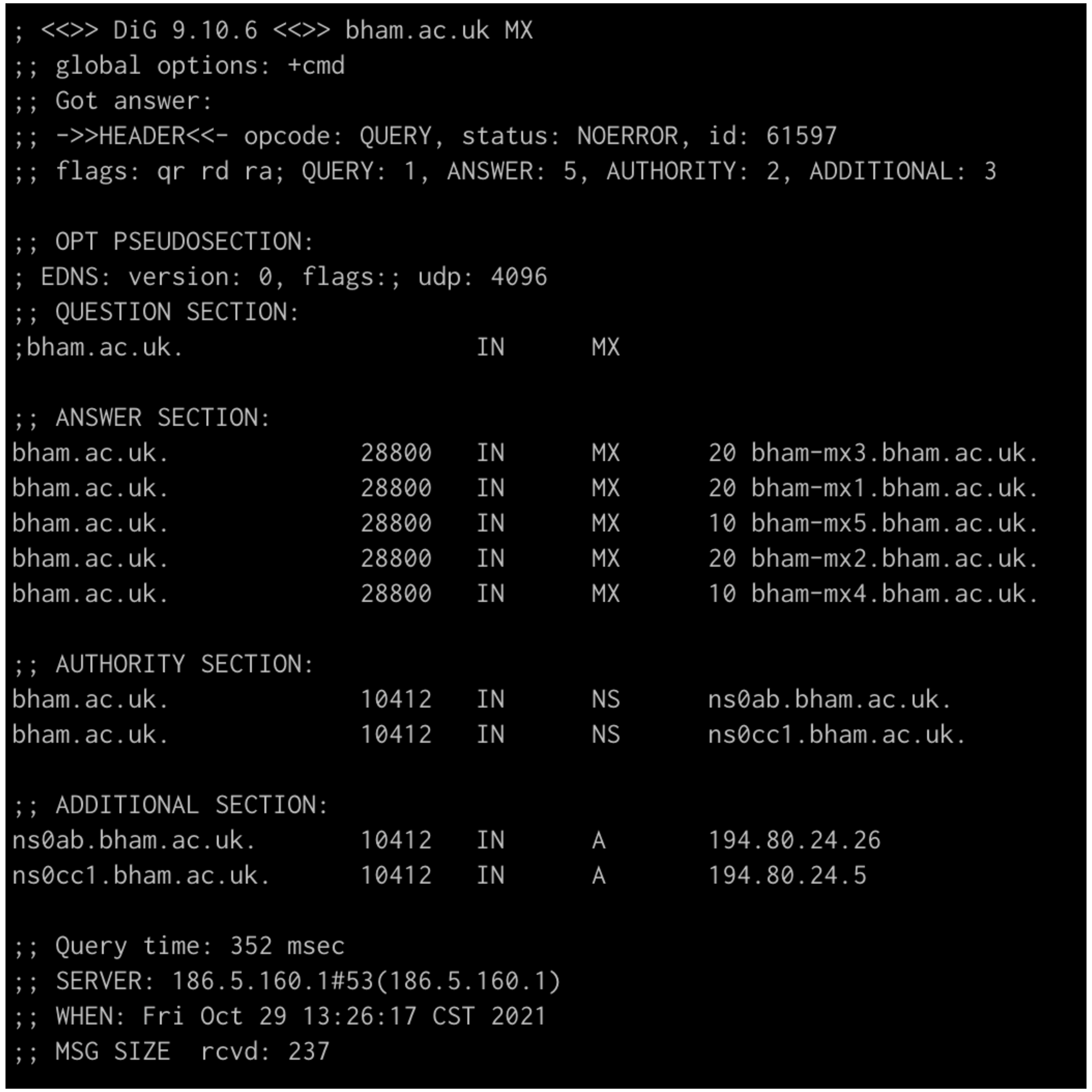
- Each DNS record has a **time-to-live** attached, indicating how long an entry can be cached.



Real-world DNS Examples



```
; <<>> DiG 9.10.6 <<>> bham.ac.uk
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 33287
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 3

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;bham.ac.uk.                    IN      A

;; ANSWER SECTION:
bham.ac.uk.             6650    IN      A       147.188.128.127

;; AUTHORITY SECTION:
bham.ac.uk.             10457   IN      NS      ns0ab.bham.ac.uk.
bham.ac.uk.             10457   IN      NS      ns0cc1.bham.ac.uk.

;; ADDITIONAL SECTION:
ns0ab.bham.ac.uk.       10457   IN      A       194.80.24.26
ns0cc1.bham.ac.uk.      10457   IN      A       194.80.24.5

;; Query time: 118 msec
;; SERVER: 186.5.160.1#53(186.5.160.1)
;; WHEN: Fri Oct 29 13:25:32 CST 2021
;; MSG SIZE  rcvd: 128
```



```
; <<>> DiG 9.10.6 <<>> bham.ac.uk MX
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 61597
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 2, ADDITIONAL: 3

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;bham.ac.uk.                    IN      MX

;; ANSWER SECTION:
bham.ac.uk.             28800   IN      MX      20 bham-mx3.bham.ac.uk.
bham.ac.uk.             28800   IN      MX      20 bham-mx1.bham.ac.uk.
bham.ac.uk.             28800   IN      MX      10 bham-mx5.bham.ac.uk.
bham.ac.uk.             28800   IN      MX      20 bham-mx2.bham.ac.uk.
bham.ac.uk.             28800   IN      MX      10 bham-mx4.bham.ac.uk.

;; AUTHORITY SECTION:
bham.ac.uk.             10412   IN      NS      ns0ab.bham.ac.uk.
bham.ac.uk.             10412   IN      NS      ns0cc1.bham.ac.uk.

;; ADDITIONAL SECTION:
ns0ab.bham.ac.uk.       10412   IN      A       194.80.24.26
ns0cc1.bham.ac.uk.      10412   IN      A       194.80.24.5

;; Query time: 352 msec
;; SERVER: 186.5.160.1#53(186.5.160.1)
;; WHEN: Fri Oct 29 13:26:17 CST 2021
;; MSG SIZE  rcvd: 237
```

DNS and the Web

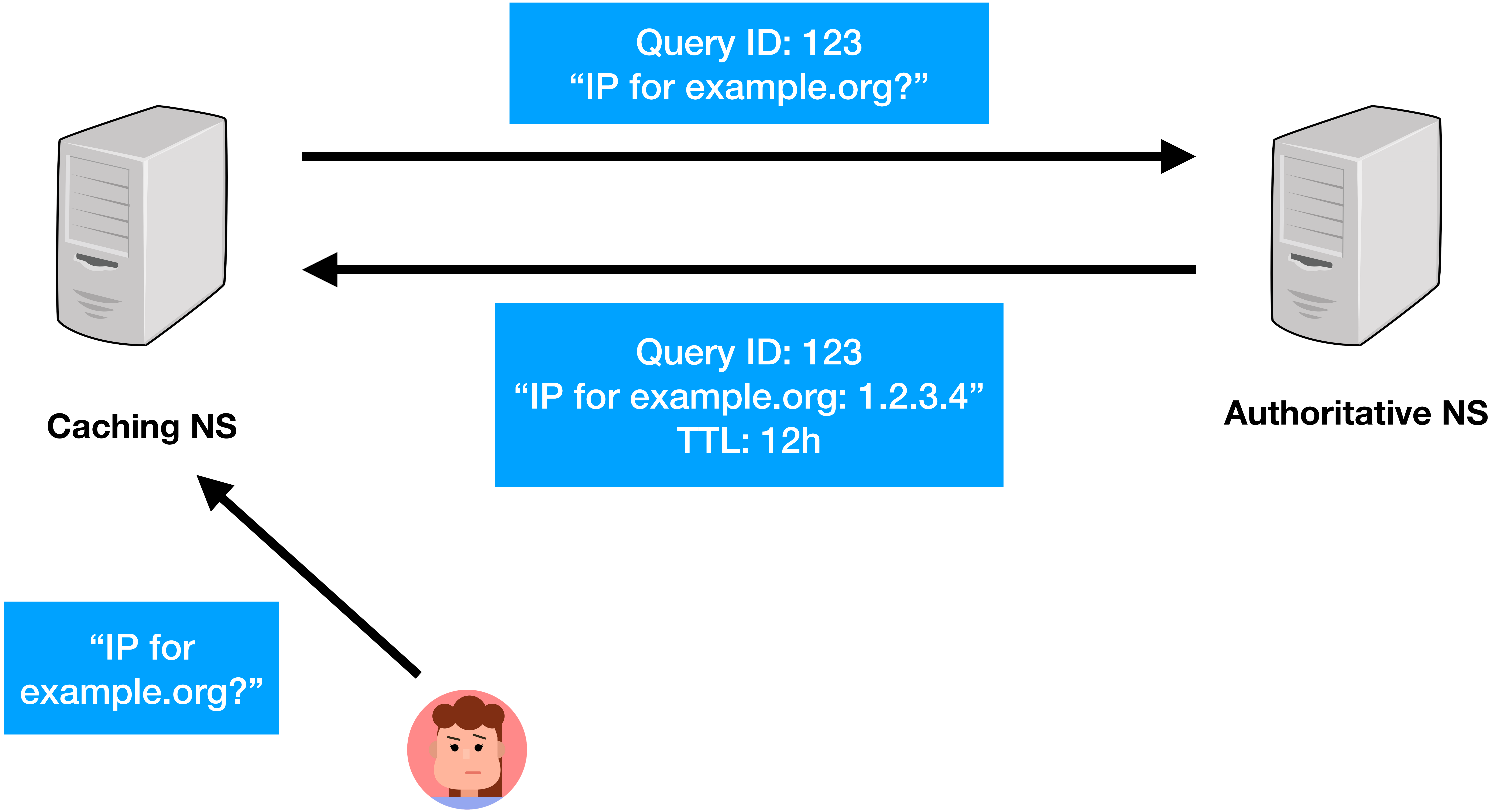
- Whenever you browse to a **webpage** (e.g., https://en.wikipedia.org/wiki/Domain_Name_System), a DNS query is made to identify the relevant **IP address**.
- Then, your browser opens a **TCP connection** with that IP address on a default port (if no explicit port is given in the URL)
    - In case of http://, the default port is 80.
    - In case of https://, the default port is 443, and the TCP connection is secured via **TLS**.
- The browser then talks to the web server using the **HTTP protocol** over this connection.

DNS Issues

- **DNS cache poisoning:** delivering wrong/malicious information to caching servers with a long time-to-live
- DNS queries and responses are sent unencrypted
    - **DNS hijacking:** subverting the resolution of DNS queries to other NS
    - **MITM attacks**
    - **Privacy:** eavesdropping on DNS queries

Cache Poisoning

- DNS Queries/Responses are sent via UDP.



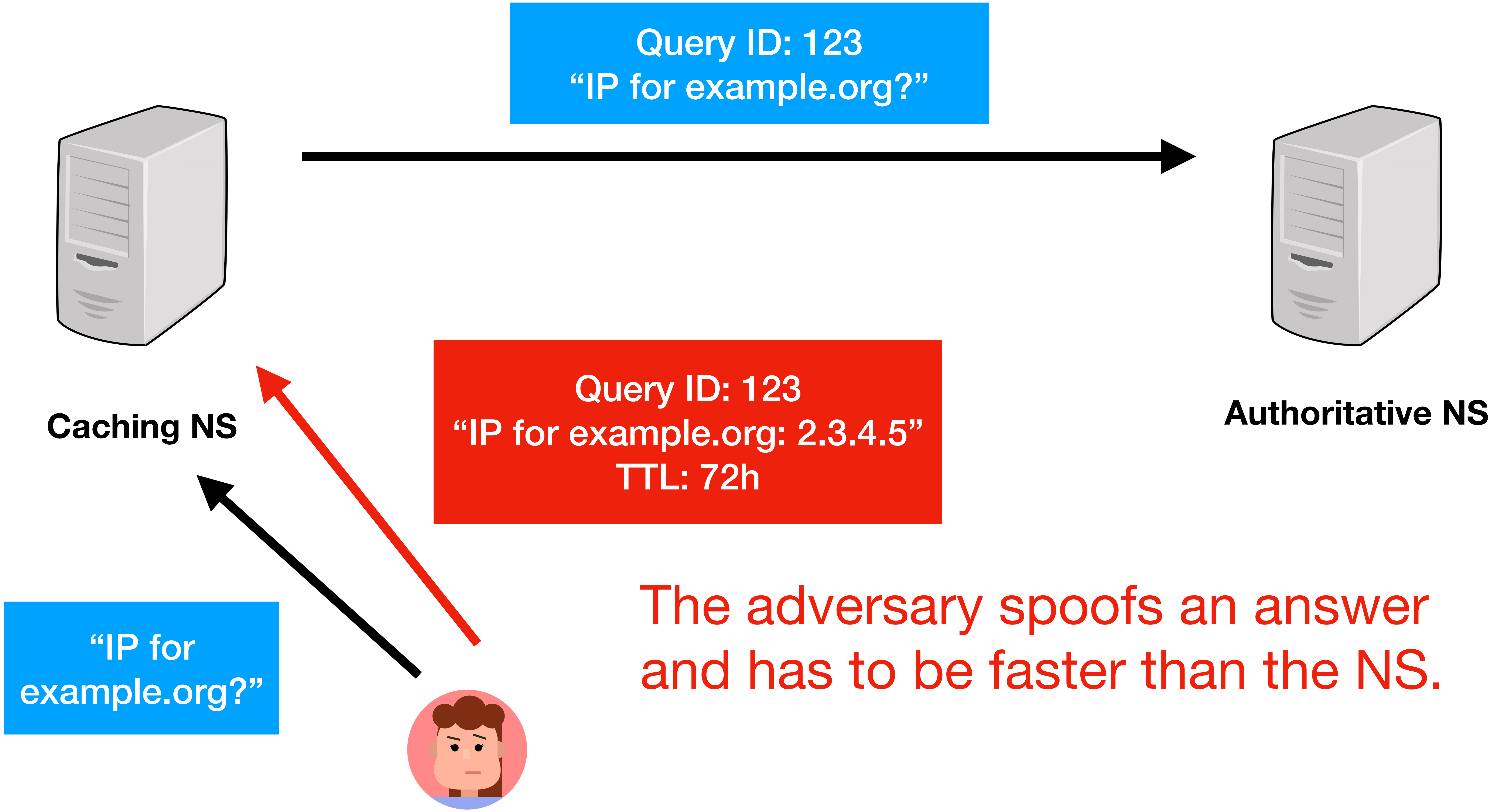
Query ID: 123
“IP for example.org?”

Query ID: 123
“IP for example.org: 1.2.3.4”
TTL: 12h

“IP for example.org?”

Cache Poisoning

- DNS Queries/Responses are sent via UDP.



Query ID: 123
“IP for example.org?”

Caching NS

Authoritative NS

Query ID: 123
“IP for example.org: 2.3.4.5”
TTL: 72h

“IP for example.org?”

The adversary spoofs an answer and has to be faster than the NS.

Cache Poisoning

- Mitigations:
    - **Random Query IDs** (sometimes implemented as incrementing),
      still allows guessing of Query IDs (only 16-bit)
    - **Randomise Source Ports** (instead of sending all queries from a fixed port);
      now the adversary has to guess the port as well to spoof the packet

DNSSEC

- DNSSec provides **authentication** (including authenticated denial of existence) and **data integrity** via digital signatures.
- It does not solve the issues regarding **confidentiality** or **availability**.
- All answers from DNSSec protected zones are signed.

DNSSEC

- DNSSec introduces additional types of resource record:
  - **DNSKEY:** Contains a public key.
  - **RRSIG:** A digital signature for other resource records.
  - **NSEC:** Authenticated denial of existence record.
  - **DS:** Delegation signer (links to a DNSKEY in a sub-NS).
- The DS and DNSKEY records create a chain of trust but still require a **trust anchor**.

How to change your keys?

- Create a second set of entries until the time-to-live should have been expired.
- Then delete the old set of keys/signatures.
- For trust anchors, such as the root entries, this is more complicated: These keys might be stored in operating systems and might require updates thereof.

Summary

- DNS operates without encryption/signatures over UDP and can be vulnerable to
  - Cache poisoning
  - MITM
  - Hijacking
  - Eavesdropping

- DNSSEC is an extension of DNS
  - It provides authentication and data integrity
  - It relies on digital signatures
  - It does not solve confidentiality or availability issues