DNS
Domain Name System

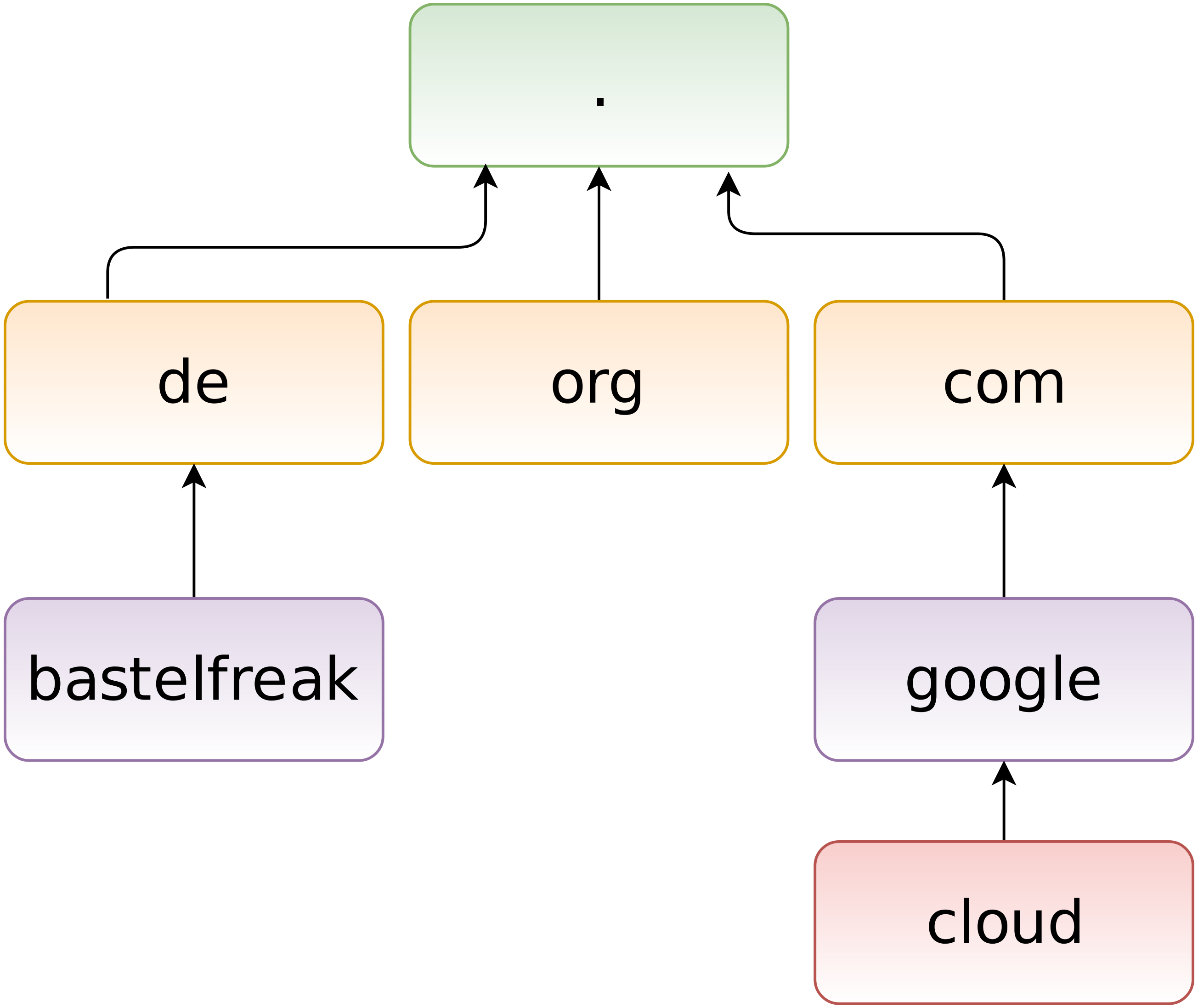
Content

- Why do we need this?
- Hierarchy
- Setup
- Zonesfile
- Bonus?

Why?

- The internet works with ip addresses
- Remembering a domain is easier than an ip address
- IPv6 addresses are even harder to remember than IPv4
- We need a solution to resolve FQDNs into ip addresses

Hierarchy



Setup

Root zone

- The root zone represents the dot in our hierarchy
- 13 anycast clusters
  - Distributed over the whole world
  - Currently over 500 physical machines serve those clusters
  - All of them support IPv4 and IPv6, most of them anycast
- Managed by the ICANN
  - Internet Corporation for Assigned Names and Numbers
- More details + map at http://root-servers.org/

Setup

Root zone

Why 13?

- In the first version, a DNS payload could carry 512 bytes
- You can fit 13 ipv4 addresses reliable into one payload
- They are named a.root-servers.net to m.root-servers.net
- Back in the days, IPv6 wasn't a thing

Setup

Root zone

Why 13?

Registries

- domain name registries are responsible for top level domains
- ccTLD are managed by national registries (DENIC for .de)
- They sell/delegate domains within their top level domains
- They maintain a list of reponsible name servers for each zone within their scope

Setup

Root zone

Why 13?

Registries

DNS provider

- Dedicated company or the registry itself
- At least one zonefile for each domain

Zonefile

Example

```
$ORIGIN bastelfreak.de.
$TTL 601
@ 300   IN SOA dns.ovh.de. hostmaster.ovh.de. (
          2017091500 ; Serial
          10800   ; Refresh 3 hours
          3600    ; Retry   1 hour
          3600000 ; Expire  1000 hours
          300 )   ; Minimum 5 mins
      IN    NS    dns.hosteurope.de.
      IN    NS    dns2.hosteurope.de.

      IN    MX    50  mx0.hosteurope.de.

www   IN    A        95.156.226.70
www   IN    AAAA  2a05:bec0:30:9::1
; eof
```

Zonefile

Example

Content

- Contains resource records
- Name - Obvious?
- TTL - Time to live, can be omitted and will then be inherited
- Record class - The namespace, "IN" in 99% of the usecases
- Record type - Defines the type of information
- Record data - the actual data

Zonefile

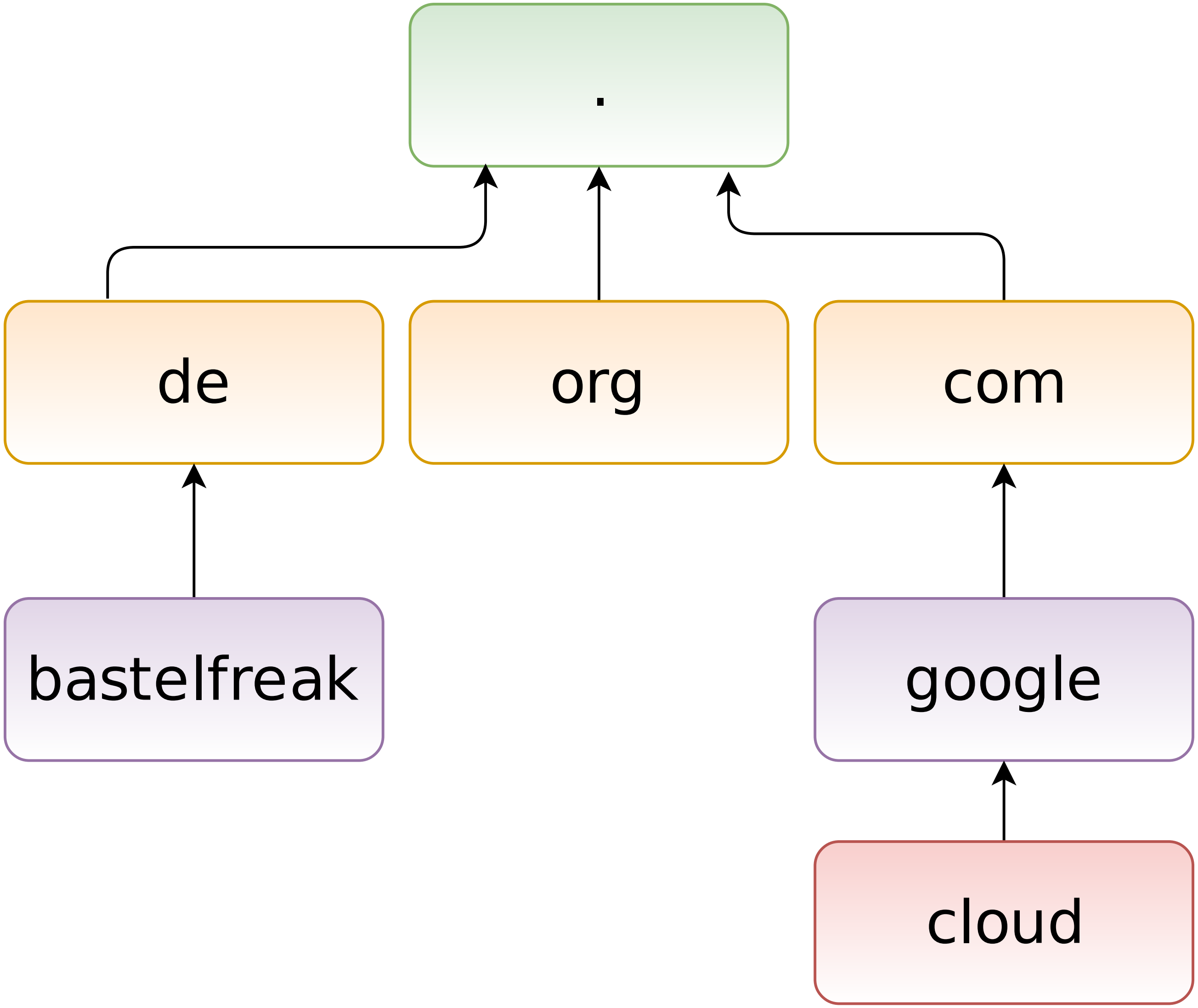
Record Types

- 96 different types are available
- A - Refers to an IPv4 address
- AAAA - Refers to an IPv6 address
- MX - Refers to the responsible SMTP server for this domain
- PTR - translates an IP address into a domain
- SRV - service locator, redirects to a specifc IP/Port where a service is running

Bonus

Bonus

DNSSEC



Bonus

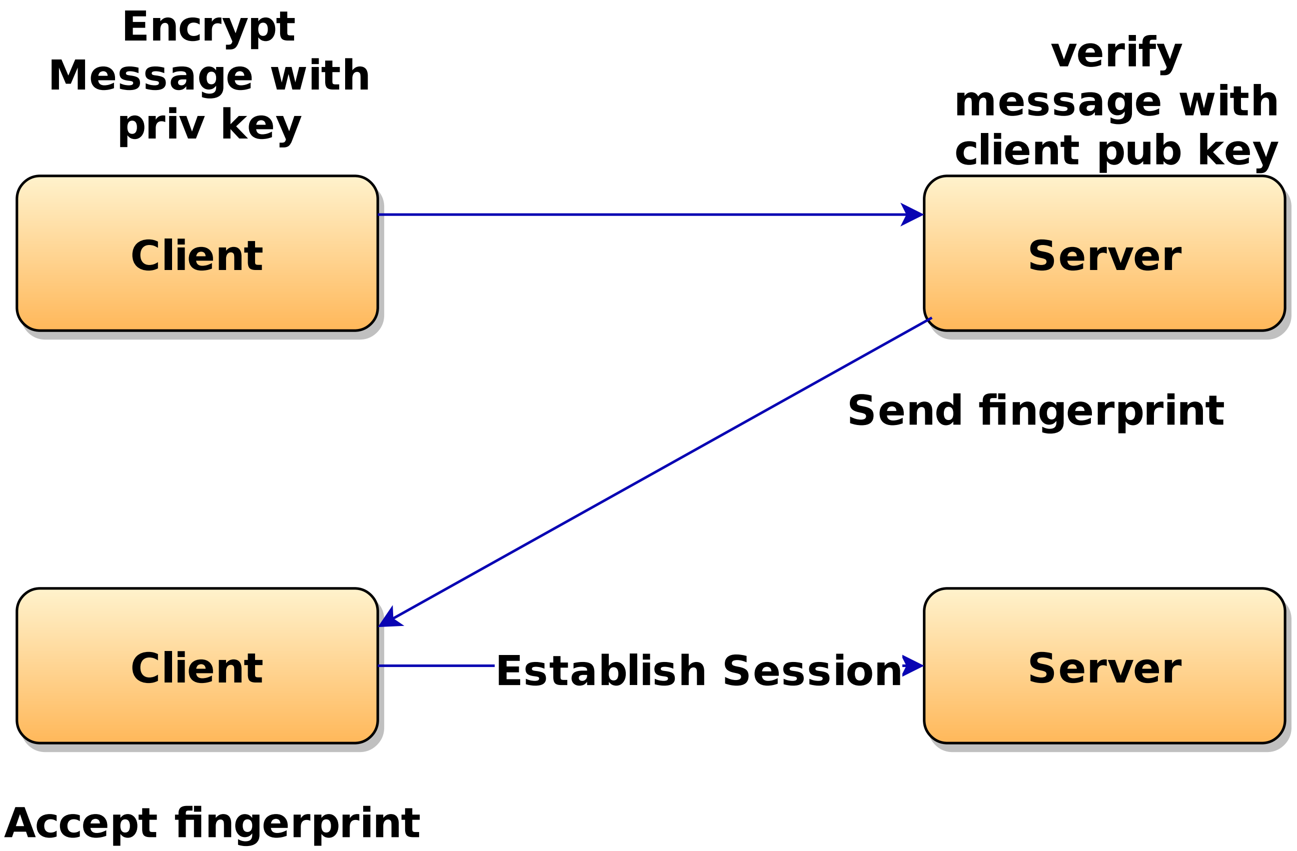
DNSSEC

```
denic.de.        86399 IN DS 26155 8 2 (
  078547BF937A225C9EFB2CAF7FAC11BD88671FCEDFAE
  EB55C9B19CB0320BEFC9 )
denic.de.   3599 IN RRSIG A 8 2 3600 (
   20170930090000 20170916090000 26155 denic.de.
   TEbgBeNqyBX1wzV8TF6GQAKwWFwK6oGl8jZEW1aOpANm
   7U5nxAWk+GWTHQSPtNQYVBOhyukGOZQheuHv202ZuSOv
   cTjmqSVjfuswtJFkU0AKW3EaEJlKduaXUmaJPtNlyBwY
   HvKY440akDeECPbUihKd03UYAEI1JZm4cmT43lV89XqY
   hfkIOnTkRywUmTPrsEC29FRz4zCT2syf7LM8IhyU9Uke
   SnshWpB+9uW3D1yo+1SmoiHWPXrJekBNhecI )
```

Bonus

DNSSEC

SSH PKI



Bonus

DNSSEC

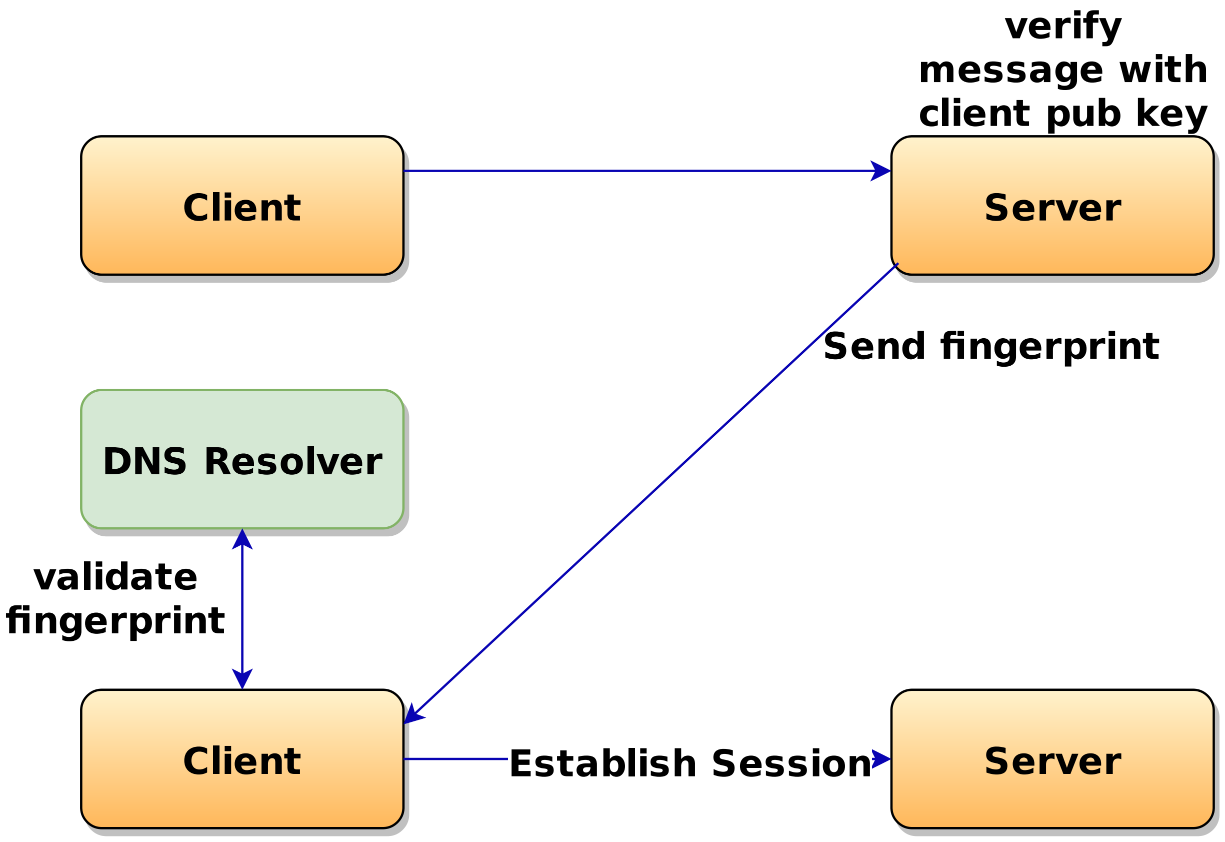
SSH PKI

- PKI = Public Key Infrastructure
- SSH keys are asymetric keys with public + private keys
- Hash of the public key from the server can be saved with the SSHFP record type in a zonefile
- During a ssh connection, a client retrieves the fingerprints and checks DNS
- Man in the middle attacks are now very hard to achieve
    - If you somehow get your box between the original server + client, you still need to adjust the keys in DNS

Bonus

DNSSEC

SSH PKI



Bonus

DNSSEC

SSH PKE

DANE

- DANE = DNS-based Authentication of Named Entities
- CAA resource record type
  - Enforces a certain CA for a domain

```
bastelfreak.de.  IN CAA 0 issue letsencrypt.org
bastelfreak.de.  IN CAA 0 iodef mailto:me@bastelfreak.de
```

Bonus

DNSSEC

SSH PKI

DANE

- TLSA record which provides a certificate fingerprint for a specific protocol + port + domain

```
_443._tcp.bastelfreak.de IN TLSA 4378568437568456856856
```

Bonus

DNSSEC

SSH PKI

DANE

Conclusion

- DNS itself is a simple but insecure protocol
- DNS can be secure with DNSSEC
- DNS is capable of storing many different kinds of information
- DNS is great to improve authentication of other protocols
- DNSSEC + DANE/SSHFP provide a trusted way of authentication