DNSSEC
All You Need To Know To Get Started

Olaf M. Kolkman

RIPE NCC

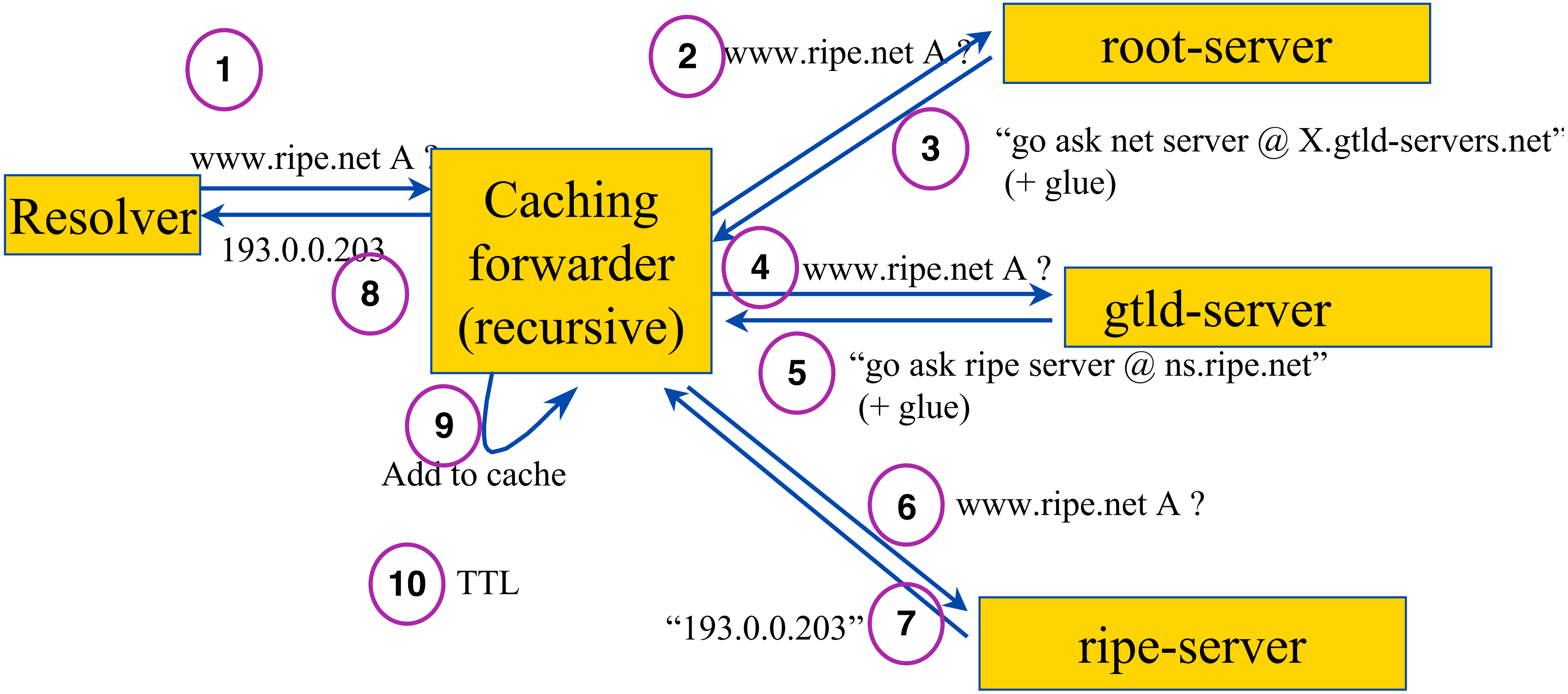
A Semi Technical Introduction

- Why do we need DNSSEC
- What does DNSSEC provide
- How does DNSSEC work

Reminder: DNS Resolving

Question:

www.ripe.net A



1

www.ripe.net A ?

Resolver

193.0.0.203

8

Caching forwarder (recursive)

2 www.ripe.net A ?

root-server

3 “go ask net server @ X.gtld-servers.net”
(+ glue)

4 www.ripe.net A ?

gtld-server

5 “go ask ripe server @ ns.ripe.net”
(+ glue)

9 Add to cache

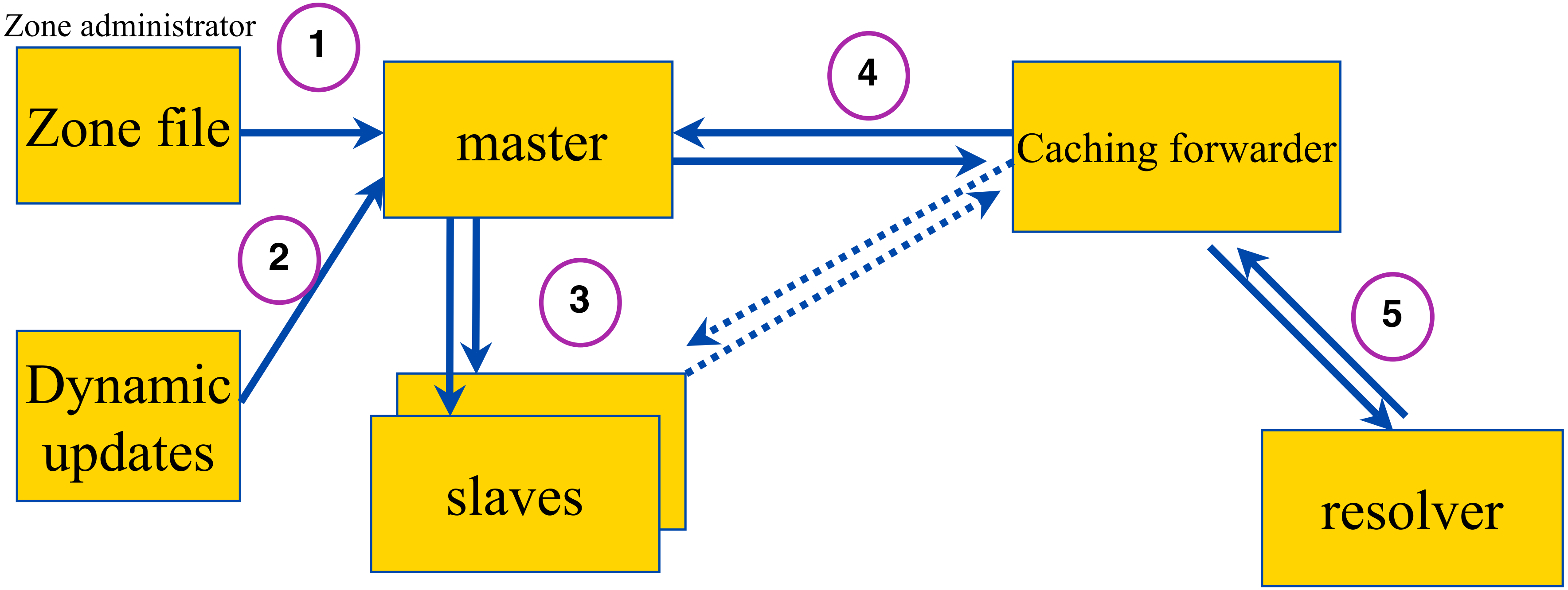
6 www.ripe.net A ?

ripe-server

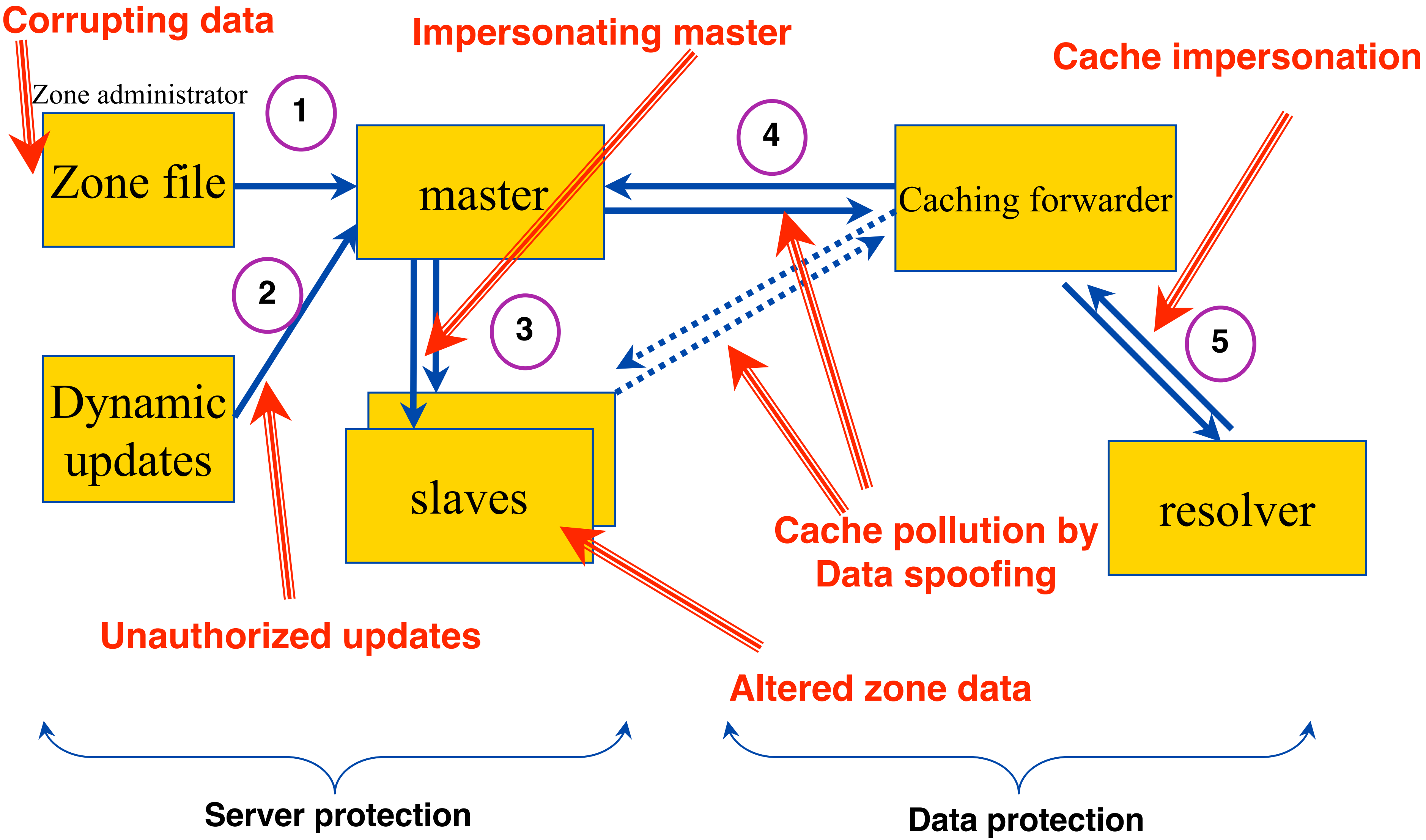
10 TTL

“193.0.0.203” 7

DNS: Data Flow



DNS Vulnerabilities



DNS Protocol Vulnerability

- DNS data can be spoofed and corrupted between master server and resolver or forwarder
- The DNS protocol does not allow you to check the validity of DNS data
    - Exploited by bugs in resolver implementation (predictable transaction ID)
    - Polluted caching forwarders can cause harm for quite some time (TTL)
    - Corrupted DNS data might end up in caches and stay there for a long time
- How does a slave (secondary) knows it is talking to the proper master (primary)?

DNSSec protects..

DNSSec protects against data spoofing and corruption

- TSIG/SIG0: provides mechanisms to authenticate communication between servers
- DNSKEY/RRSIG/NSEC: provides mechanisms to establish authenticity and integrity of data
- DS: provides a mechanism to delegate trust to public keys of third parties

- A secure DNS will be used as an infrastructure with public keys
  - However it is **NOT** a PKI

Core Elements (1)

DNSSEC is based on Public Key Cryptography

- Key pair: a private and a public key
- The private key can be used to create signatures
- The signature can be ‘validated’ with the public key.
- If the signature over a message validates the message must have been signed by the holder of the private keys.
- The message is *not* encrypted

Core elements 2

- Public Key Crypto is about private keys, public keys and signatures.
- Also about building and validating chains of trust

- Public keys are published in the DNS
- Signatures made over the data is published in the DNS
- Chains of trust are build from parent to child
- How about those private keys?

In Practice (Signatures)

- Using the private key of a keypair a zonesigner adds signatures to RR sets.

```
tld.          100      IN SOA  ns.registry.TLD. olaf.ripe.net. (
                        2002050501 100 200 604800 100 )

tld.           100      RRSIG   SOA 1 1 100 20040718114001 (
                              20040618114001 37958 tld.
                              uTTqESj2D65OZ7a4Q2ruGZwsmlGoeiDbnzbD
                              X0WMjkhY0IK2kifw5xDYViYHFtfvZIlKeV9M
                              VEW9m6L5uJubi9zBZwAI8xSln8UWO6NuhXxc
                              MsOUEsxm9sVh5HbZOjQC6XOI9Um1gOCMABW3
                              O/jZf5gon3UxVt9YRbzZuYD0pRg= )
```

In Practice (Keys)

The DNSKEY RR is published at the apex of the zone.

(*apex is the beginning, the start, where the SOA RR lives*)

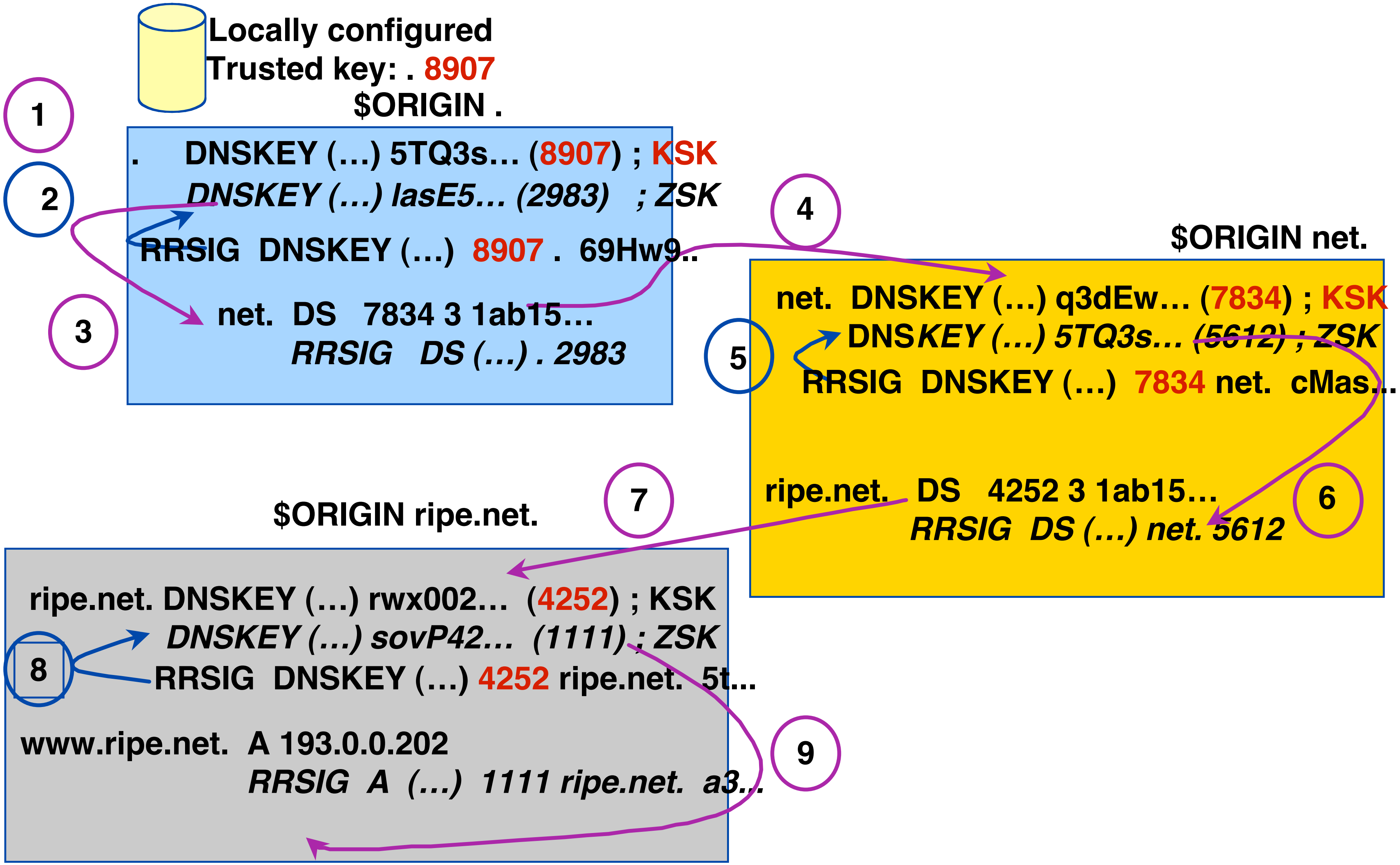
```
tld. IN DNSKEY 256 3 1 (
  AQPQOhIjhTLvcDjo9xQJN0Z0Tj33UmvxJlb85CbgB+7PlqDnh0hZwoZo
  OigR2fYYbmdIr/Oj+HzKy8sM9Jwsghv6FWYEIMeQR2IyeMiZ6sho93ID
  7Rm8cG07yVHARTWzXdLx2zi2Hj6yDPn1asL4TTvXamocjM6IJqaWgEMN
  SpRG7Q== )
```

In Practice (Chain of trust)

- Data from a zone can be verified using the DNSKEY from the same zone.
- For each zone you want to verify the data from you will need a DNSKEY.
- Use the DNS to build chains of trust
  - Just like NS tells one where the nameservers for the child zone are
  - The DS tells one where the DNSKEYs for the zonefile can be found.

DS is a pointer to the next key in the chain of trust.

Walking the Chain of Trust



Locally configured
Trusted key: . 8907

$ORIGIN .

. DNSKEY (…) 5TQ3s… (8907) ; KSK
DNSKEY (…) lasE5… (2983) ; ZSK
RRSIG DNSKEY (…) 8907 . 69Hw9..

net. DS 7834 3 1ab15…
RRSIG DS (…) . 2983

$ORIGIN net.

net. DNSKEY (…) q3dEw… (7834) ; KSK
DNS*KEY (…) 5TQ3s… (5612) ; ZSK*
RRSIG DNSKEY (…) 7834 net. cMas...

ripe.net. DS 4252 3 1ab15…
RRSIG DS (…) net. 5612

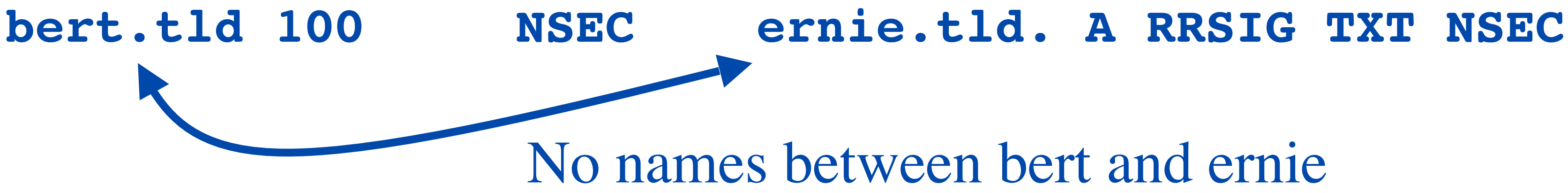
$ORIGIN ripe.net.

ripe.net. DNSKEY (…) rwx002… (4252) ; KSK
DNSKEY (…) sovP42… (1111) ; ZSK
RRSIG DNSKEY (…) 4252 ripe.net. 5t…

www.ripe.net. A 193.0.0.202
RRSIG A (…) 1111 ripe.net. a3...

But what if data is not in the DNS

- NSEC RR is used to proof non-existence of data.
- It tells us
  - which names cannot be found in the DNS
  - and which types are not available in the DNS



```
bert.tld 100     NSEC     ernie.tld. A RRSIG TXT NSEC
```

No names between bert and ernie

- Zone enumeration problem.

A NSEC B, B NSEC P, P NSEC Q, Q NSEC A

DNSSec Current State

- Changes to the specs that are now going through the IETF.
  - The last hurdles are being taken
- Various people are trying to drive deployment.
  - RIPE NCC provides a course, develops tools, is involved in development of procedures and strives for early deployment.
- Zone enumeration problem will be studied by the IETF after DNSSEC has been standarised

Questions???

- Questions and feedback to olaf@ripe.net